
NIST Technical Note 2225

Economic Evaluation Engine (E3) Technical Manual

Joshua Kneifel
David Webb
Luke Donmoyer
Tasha Kim

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.TN.2225>



NIST
**National Institute of
Standards and Technology**
U.S. Department of Commerce

NIST Technical Note 2225

Economic Evaluation Engine (E3) Technical Manual

Joshua Kneifel
David Webb
Luke Donmoyer
Tasha Kim
Engineering Laboratory

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.TN.2225>

May 2022



U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology

Certain commercial entities, equipment, or materials may be identified in this document to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

NIST-developed software is expressly provided "AS IS." NIST MAKES NO WARRANTY OF ANY KIND, EXPRESS, IMPLIED, IN FACT OR ARISING BY OPERATION OF LAW, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT AND DATA ACCURACY. NIST NEITHER REPRESENTS NOR WARRANTS THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT ANY DEFECTS WILL BE CORRECTED. NIST DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE OF THE SOFTWARE OR THE RESULTS THEREOF, INCLUDING BUT NOT LIMITED TO THE CORRECTNESS, ACCURACY, RELIABILITY, OR USEFULNESS OF THE SOFTWARE.

National Institute of Standards and Technology Technical Note 2225
Natl. Inst. Stand. Technol. Tech. Note 2225, 142 pages (May 2022)
CODEN: NTNOEF

This publication is available free of charge from:
[https:// doi.org/10.6028/NIST.TN.2225](https://doi.org/10.6028/NIST.TN.2225)

Abstract

The Engineering Laboratory (EL) of the National Institute of Standards and Technology (NIST) has developed a variety of tools that provide some level of economic analysis, many of which have been developed by EL's Applied Economics Office (AEO). AEO-developed tools range across topic areas and economic metrics. Each tool is targeting government agencies and organizations that fall under NIST's mission. Stakeholders need different levels of details and functionality because of the diverse backgrounds across stakeholders. As a result, AEO has developed these software tools in a "one-off" approach on an as needed basis. As a result, each software has its own unique set of code even though these tools have some key underlying components that are common across tools, including methodology documented in standards.

To improve the efficiency of the tools developed by AEO and its stakeholders, AEO has developed an economic service/engine with an associated Application Programming Interface (API) that provides consistent, standardized, transparent, reproducible economic analysis calculations should be developed, maintained, and made publicly available for internal and external researchers, analysts, and software developers called the Economic Evaluation Engine (E3). E3 provides economic evaluation capabilities based on ASTM building economic standards including life-cycle cost analysis, benefit-cost analysis, profit maximization / cost-minimization, and risk-cost analysis. E3 could be leveraged by NIST as well as other federal agencies and national laboratories, state and local governments, academia, and the private sector to improve their tool development by using a federally developed and validated calculations to provide standard-consistent analysis.

E3 is publicly accessible on a NIST server as well as on GitHub for anyone to use for economic analysis, whether it's through a basic script calling on the E3 API, online interface that connects to the E3 API, or an executable program that is built on E3's capabilities. Additionally, users of E3 could provide expansions to the capabilities by developing the code and submitting it to NIST for review and incorporation.

This technical manual describes E3, including the underlying standards on which its based, the data flow through the tool, details on the code libraries, classes, and file structure, development tools used, guidance on how to use the tool, limitations, planned release, and the development of its future capabilities.

Keywords

Building economics; life-cycle cost analysis; benefit-cost analysis; sustainability; resilience; software; building standards

Preface

This documentation was developed by the Applied Economics Office (AEO) in the Engineering Laboratory (EL) at the National Institute of Standards and Technology (NIST). The document explains how the E3 software has been developed. The intended audience are researchers and software developers completing economic analysis, such as life cycle cost analysis and benefit-cost analysis.

Disclaimers

The policy of the National Institute of Standards and Technology is to use metric units in all its published materials. Because this report is intended for the U.S. construction industry which uses U.S. customary units, it is more practical and less confusing to include U.S. customary units as well as metric units. Measurement values in this report are therefore stated in metric units first, followed by the corresponding values in U.S. customary units within parentheses.

NIST-developed software is expressly provided "AS IS." NIST MAKES NO WARRANTY OF ANY KIND, EXPRESS, IMPLIED, IN FACT OR ARISING BY OPERATION OF LAW, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT AND DATA ACCURACY. NIST NEITHER REPRESENTS NOR WARRANTS THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT ANY DEFECTS WILL BE CORRECTED. NIST DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE OF THE SOFTWARE OR THE RESULTS THEREOF, INCLUDING BUT NOT LIMITED TO THE CORRECTNESS, ACCURACY, RELIABILITY, OR USEFULNESS OF THE SOFTWARE.

Acknowledgements

The authors wish to thank all those who contributed ideas and suggestions for this report. They include Dr. Jennifer Helgeson and Dr. David Butry of EL's Applied Economics Office, Priya Lavappa of EL Data, Security, and Technology (ELDST), and Dr. Nicos S. Martys of EL's Materials and Structural Systems Division.

Author Information

Joshua Kneifel
Economist
National Institute of Standards and Technology
Engineering Laboratory
100 Bureau Drive, Mailstop 8603
Gaithersburg, MD 20899 8603
Email: joshua.kneifel@nist.gov

David Webb
Engineer
National Institute of Standards and Technology
Engineering Laboratory
100 Bureau Drive, Mailstop 8603
Gaithersburg, MD 20899 8603
Email: david.webb@nist.gov

Luke Donmoyer
PREP Research Associate
National Institute of Standards and Technology
Engineering Laboratory
100 Bureau Drive, Mailstop 8603
Gaithersburg, MD 20899 8603
Email: luke.donmoyer@nist.gov

Tasha Kim
PREP Research Associate
National Institute of Standards and Technology
Engineering Laboratory
100 Bureau Drive, Mailstop 8603
Gaithersburg, MD 20899 8603
Email: jaeyoung.kim@nist.gov

Contents

ABSTRACT	I
PREFACE.....	II
ACKNOWLEDGEMENTS.....	III
AUTHOR INFORMATION	IV
LIST OF ACRONYMS	XI
1 INTRODUCTION AND BACKGROUND.....	1
1.1 AEO'S SOFTWARE TOOL DEVELOPMENT	1
1.2 ECONOMIC EVALUATION ENGINE (E3)	3
1.2.1 Concept.....	4
1.2.2 Progress/Status.....	6
1.3 ASTM STANDARDS-BASED METHODOLOGY.....	7
1.3.1 Building Economics (ASTM E06.81)	7
1.3.1.1 LCCA and BCA.....	7
1.3.1.2 Uncertainty and Risk	14
1.3.1.3 Analytical Hierarchy Process (AHP)	15
1.3.1.4 Applicability of Building Economics Standards.....	15
1.3.2 Community Resilience (E06.81)	16
1.3.3 Sustainable Manufacturing (E60.13).....	16
2 DATA FLOW THROUGH THE API.....	19
2.1 DIRECTORY STRUCTURE FOR FILES MENTIONED IN FIGURE 2-2	20
2.2 INPUT CLASSES (USER-DEFINED CLASSES).....	20
2.2.1 Analysis (AnalysisSerializer).....	20
2.2.2 Alternative (AlternativeSerializer)	21
2.2.3 Benefit / Cost / Non-monetary (BcnSerializer).....	21
2.2.4 Sensitivity (SensitivitySerializer).....	21
2.2.5 Input (InputSerializer)	21
2.3 LIBRARIES	21
2.3.1 Support Libraries.....	21
2.3.1.1 tasks	22
2.3.1.2 registry	22
2.3.2 Modules	22
2.3.2.1 cashflow	22
2.3.2.2 measures.....	22
2.3.2.3 optional	23
2.3.2.4 required	23
2.3.2.5 sensitivity	23
2.3.2.6 Module Workflow	23
2.3.3 Output Classes	23
2.3.3.1 CashFlow (CashFlowSerializer)	24
2.3.3.2 AlternativeSummary (AlternativeSummarySerializer)	24
2.3.3.3 SensitivitySummary (SensitivitySummarySerializer)	24
2.3.3.4 Output (OutputSerializer)	24
2.4 DATA FLOW THROUGH THE API	24
3 CODE, LIBRARY, AND CLASS DETAILS	27

3.1	DEVELOPMENT TOOLS	27
3.1.1	<i>Programming – Python, numPy, and pytest</i>	27
3.1.2	<i>Framework – Django, Django REST Framework, and Django.test</i>	27
3.1.3	<i>Deployment – PostgreSQL, Docker, AWS</i>	28
3.1.4	<i>Input/Output Format</i>	29
3.2	E3 FILES	29
3.2.1	<i>Input Classes (User-Input Classes)</i>	29
3.2.1.1	Analysis (AnalysisSerializer)	29
3.2.1.1.1	Functions	30
3.2.1.1.2	Classes	30
3.2.1.2	Alternative (AlternativeSerializer)	31
3.2.1.2.1	Functions	31
3.2.1.2.2	Classes	32
3.2.1.3	BCN (BcnSerializer)	32
3.2.1.3.1	Functions	32
3.2.1.3.2	Classes	32
3.2.1.4	Sensitivity (SensitivitySerializer)	34
3.2.1.4.1	Functions	34
3.2.1.4.2	Classes	34
3.2.1.5	Input (InputSerializer)	35
3.2.1.5.1	Functions	36
3.2.1.5.2	Classes	36
3.2.2	<i>API Defined Classes</i>	36
3.2.2.1	AlternativeSummary (AlternativeSummarySerializer)	36
3.2.2.1.1	Functions	36
3.2.2.1.2	Classes	40
3.2.2.2	CashFlow (CashFlowSerializer)	42
3.2.2.2.1	Functions	42
3.2.2.2.2	Classes	42
3.2.2.3	SensitivitySummary (SensitivitySummarySerializer)	46
3.2.2.3.1	Functions	46
3.2.2.3.2	Classes	46
3.2.2.4	Output (OutputSerializer)	49
3.2.2.4.1	Functions	49
3.2.2.4.2	Classes	49
3.2.3	<i>Modules</i>	49
3.2.3.1	cashflows	50
3.2.3.1.1	Functions	50
3.2.3.1.2	Classes	52
3.2.3.2	measures	52
3.2.3.2.1	Functions	52
3.2.3.2.2	Classes	52
3.2.3.3	optional	53
3.2.3.3.1	Functions	53
3.2.3.3.2	Classes	53
3.2.3.4	required	54
3.2.3.4.1	Functions	54
3.2.3.4.2	Classes	54
3.2.3.5	sensitivity	55
3.2.3.5.1	Functions	55
3.2.3.5.2	Classes	55
3.2.4	<i>Support Libraries – Calculations</i>	55
3.2.4.1	tasks	56

3.2.4.1.1	Functions.....	56
3.2.4.1.2	Classes.....	56
3.2.4.2	registry.....	56
3.2.4.2.1	Functions.....	56
3.2.4.2.2	Classes.....	56
3.2.4.3	Variables.....	58
3.2.4.3.1	Functions.....	58
3.2.4.3.2	Classes.....	58
3.2.5	<i>Support Libraries – API</i>	58
3.2.5.1	/API.....	59
3.2.5.2	/e3_django.....	59
3.2.5.3	/E3_local.....	59
3.2.5.4	/frontend.....	59
3.2.6	<i>Additional Files</i>	59
3.3	PLANNED CHANGES IN DEVELOPMENT TOOLS AND FILE STRUCTURE.....	60
3.3.1	<i>Java</i>	61
3.3.2	<i>JavaScript Submission Library</i>	61
4	USING E3	63
4.1	OVERVIEW.....	63
4.1.1	<i>Information Exchange Formats</i>	63
4.2	SENDING A REQUEST TO E3 API WEB SERVER.....	63
4.3	RUNNING E3 API LOCALLY.....	64
4.4	GUIDANCE.....	65
4.4.1	<i>General Guidance</i>	66
4.4.2	<i>Guidance by Analysis Type</i>	67
4.4.2.1	Guidance for LCCA.....	67
4.4.2.1.1	Inputs – All Real Values.....	67
4.4.2.1.2	Inputs – Combination of Real and Nominal Values.....	71
4.4.2.1.3	Inputs – Nominal Values.....	72
4.4.2.2	Guidance for BCA.....	72
4.4.2.3	Guidance for Profit Maximization.....	75
4.4.2.4	Guidance for Monetary and Non-monetary Synergies and Trade-Offs.....	77
4.4.3	<i>Other Output Results Options</i>	79
4.4.4	<i>Sensitivity and Uncertainty</i>	79
4.4.4.1	Sensitivity.....	79
4.4.4.2	Uncertainty.....	80
4.4.5	<i>Edge Cases</i>	80
4.4.5.1	E3 Inputs.....	80
4.4.5.2	E3 Outputs.....	80
4.5	E3 WEB API EXAMPLES.....	81
4.5.1	<i>BLCC</i>	81
4.5.1.1	End-of-Year Discounting.....	84
4.5.1.1.1	FEMP Analysis - Energy Project.....	84
4.5.1.1.2	Federal Analysis – Financed Project.....	89
4.5.1.2	Mid-Year Discounting.....	94
4.5.1.2.1	OMB Analysis – Non-Energy Project.....	94
4.5.1.2.2	MILCON Analysis - ECIP Project.....	98
4.5.2	<i>Economic Decision Guide Software (EDGE\$)</i>	102
4.5.2.1	Continuous Discounting.....	103
4.5.2.1.1	Hurricane Resilience.....	103
4.5.2.1.2	Riverbend Flooding.....	108

4.5.2.1.3	Wildland Urban Interface (WUI).....	113
4.5.2.2	Continuous Discounting with Sensitivity.....	117
4.5.2.3	Continuous Discounting with Uncertainty	118
4.5.3	<i>Sustainable Manufacturing</i>	118
4.5.4	$[PV]^2$	122
4.5.5	<i>Other Examples</i>	122
5	PLANNED RELEASE AND CURRENT CAPABILITIES AND FUTURE CAPABILITIES.....	124
5.1	RELEASE OF E3	124
5.2	CURRENT CAPABILITIES	124
5.3	FUTURE CAPABILITIES	125
	REFERENCES	126

List of Figures

FIGURE 1-1 BLOCK DIAGRAM OF E3 PRODUCT INTERCONNECTIONS	5
FIGURE 2-2 E3 INFORMATION WORKFLOW	19

List of Tables

TABLE 1-1 AEO-DEVELOPED SOFTWARE	1
TABLE 3-2 ANALYSIS VARIABLE DETAILS	31
TABLE 3-3 ALTERNATIVE VARIABLE DETAILS.....	32
TABLE 3-4 BCN VARIABLE DETAILS	33
TABLE 3-5 SENSITIVITY OBJECT DETAILS	35
TABLE 3-6 INPUT OBJECT DETAILS	36
TABLE 3-7 ALTERNATIVE SUMMARY VARIABLE DETAILS	41
TABLE 3-8 REQUIRED CASH FLOW VARIABLE DETAILS.....	43
TABLE 3-9 OPTIONAL CASH FLOW VARIABLE DETAILS.....	46
TABLE 3-10 SENSITIVITYSUMMARY VARIABLE DETAILS.....	48
TABLE 4-11 ANALYSIS VALUES	68
TABLE 4-12 EXAMPLE COSTS.....	68
TABLE 4-13 BCN OBJECT VALUES - CONVENTIONAL	69
TABLE 4-14 BCN OBJECT VALUES – HIGH-EFFICIENCY	70
TABLE 4-15 LCCA EXAMPLE RESULTS	70
TABLE 4-16 ALTERNATIVE SUMMARY OBJECT RESULTS VALUES	71
TABLE 4-17 BCN LOAN OBJECT VALUES	72
TABLE 4-18 ANALYSIS VALUES	73
TABLE 4-19 EXAMPLE BENEFITS AND COSTS	74
TABLE 4-20 BCN OBJECT VALUES - CONVENTIONAL	74
TABLE 4-21 BCA EXAMPLE RESULTS	75
TABLE 4-22 ALTERNATIVE SUMMARY OBJECT RESULTS VALUES	75
TABLE 4-23 ANALYSIS VALUES	76
TABLE 4-24 EXAMPLE REVENUE AND COSTS	76
TABLE 4-25 BCN OBJECT VALUES – PROFIT MAXIMIZATION USING LCCA.....	77
TABLE 4-26 PROFIT MAXIMIZATION EXAMPLE RESULTS	77
TABLE 4-27 BCN OBJECT VALUES – ENVIRONMENTAL IMPACTS	78
TABLE 4-28 REAL ELECTRICITY PRICE ESCALATION RATES (YEAR-OVER-YEAR PERCENTAGE CHANGES)	84
TABLE 4-29 ANALYSIS OBJECT VALUES	85
TABLE 4-30 BCN OBJECT VALUES – EXISTING SYSTEM.....	86
TABLE 4-31 BCN OBJECT VALUES – NEW SYSTEM	87
TABLE 4-32 CASH FLOW COMPARISON	88
TABLE 4-33 SUMMARY RESULTS COMPARISON – FEMP ENERGY PROJECT.....	88
TABLE 4-34 NOMINAL ELECTRICITY PRICE ESCALATION RATES (YEAR-OVER-YEAR PERCENTAGE CHANGES)	89
TABLE 4-35 ANALYSIS OBJECT VALUES	90
TABLE 4-36 BCN OBJECT VALUES – EXISTING SYSTEM.....	91
TABLE 4-37 BCN OBJECT VALUES – NEW SYSTEM	92
TABLE 4-38 CASH FLOW COMPARISON	93
TABLE 4-39 SUMMARY RESULTS COMPARISON – FEDERAL FINANCED ENERGY PROJECT	93
TABLE 4-40 ANALYSIS OBJECT VALUES	95
TABLE 4-41 BCN OBJECT VALUES	96

TABLE 4-42 CASH FLOW COMPARISON	97
TABLE 4-43 SUMMARY RESULTS COMPARISON – OMB NON-ENERGY PROJECT	97
TABLE 4-44 REAL ELECTRICITY PRICE ESCALATION RATES (YEAR-OVER-YEAR PERCENTAGE CHANGES)	98
TABLE 4-45 ANALYSIS OBJECT VALUES	99
TABLE 4-46 BCN OBJECT VALUES	101
TABLE 4-47 DISCOUNTED SAVINGS COMPARISON	102
TABLE 4-48 SUMMARY RESULTS COMPARISON – MILCON ECIP PROJECT	102
TABLE 4-49 ANALYSIS OBJECT VALUES	104
TABLE 4-50 BENEFITS AND COSTS BY ALTERNATIVE.....	105
TABLE 4-51 BCN OBJECT VALUES – BUYOUT	106
TABLE 4-52 BCN OBJECT VALUES – LEVEE	107
TABLE 4-53 SUMMARY RESULTS COMPARISON – HURRICANE RESILIENCE	108
TABLE 4-54 ANALYSIS OBJECT VALUES	109
TABLE 4-55 BENEFITS AND COSTS BY ALTERNATIVE.....	110
TABLE 4-56 BCN OBJECT VALUES – RETROFIT	111
TABLE 4-57 BCN OBJECT VALUES – NEW BRIDGE.....	112
TABLE 4-58 SUMMARY RESULTS COMPARISON – RIVERBEND FLOODING.....	113
TABLE 4-59 ANALYSIS OBJECT VALUES	114
TABLE 4-60 BENEFITS AND COSTS BY ALTERNATIVE.....	115
TABLE 4-61 BCN OBJECT VALUES – MITIGATION	116
TABLE 4-62 SUMMARY RESULTS COMPARISON - WUI.....	117
TABLE 4-63 SUMMARY RESULTS COMPARISON – RIVERBEND FLOODING WITH SENSITIVITY – DISCOUNT RATE (GLOBAL VARIABLE)	117
TABLE 4-64 SUMMARY RESULTS COMPARISON – RIVERBEND FLOODING WITH SENSITIVITY – NEW BRIDGE CONSTRUCTION COSTS.....	118
TABLE 4-65 COSTS AND ENVIRONMENTAL IMPACTS.....	119
TABLE 4-66 BCN OBJECT VALUES – ALTERNATIVE 0 AND ALTERNATIVE 1.....	120
TABLE 4-67 BCN OBJECT VALUES – ALTERNATIVE 2 AND ALTERNATIVE 3.....	120
TABLE 4-68 SUMMARY RESULTS COMPARISON	121

List of Acronyms

Acronym	Definition
AC	air conditioning
AEO	Applied Economics Office
AHP	Analytical Hierarchy Process
AIRR	adjusted internal rate of return
API	application programming interface
ASTM	American Society for Testing and Materials
AWS	Amazon Web Services
BCA	benefit-cost analysis
BCN	benefit / cost / non-Monetary
BCR	benefit-cost ratio
BEES	Building for Environmental and Economic Sustainability
BIRDS	Building Industry Reporting and Design for Sustainability
BLCC	Building Life Cycle Cost
CFR	Code of Federal Regulations
CLI	command-line interface
DHS	Department of Homeland Security
DOE	Department of Energy
DPP	discounted payback period
DX	direct expansion
E3	Economic Evaluation Engine
ECIP	Energy Conservation Investment Program
EDG	Economic Decision Guide
EDGe\$	Economic Decision Guide Software
EO	Executive Order
EL	Engineering Laboratory
ELDST	Engineering Laboratory Data, Security, and Technology
FEMA	Federal Emergency Management Agency
FEMP	Federal Energy Management Program
FHWA	Federal Highway Administration
FRP	function reactive programing
GHG	greenhouse gas
GSA	General Services Administration
GUI	graphical user interface
HVAC	conventional heating, ventilation, and air conditioning
IRR	internal rate of return
JSON	JavaScript Object Notation

Acronym	Definition
JVM	Java Virtual Machine
LCC	life-cycle cost
LCCA	life-cycle cost analysis
LCIA	life cycle impact assessment
MARR	minimum alternative rate of return
MEER	maximum environmental expenditure rate
MILCON	military construction
NB	net benefits
NIST	National Institute of Standards and Technology
NPV	net present value
NPVE	net present value elasticity
NPVP	net present value per percent change
NS	net savings
OM	operation and maintenance
OMB	Office of Management and Budget
OMR	operations, maintenance, and repair
ORM	object-relational mapping
PV	photovoltaic
[PV]2	Present Value of PhotoVoltaics
RDF	resource description framework
REST	representational state transfer
RSS	really simple syndication
RV	residual value
SIR	savings-to-investment ratio
SPP	simple payback period
SpUD	Sprinkler Use Decisioning
SQL	Structured Query Language
ToGS	Turnout Gear Selector
TOPS-BC	Tool for Operations Benefit-Cost Analysis
WUI	wildland-urban interface

1 Introduction and Background

1.1 AEO's Software Tool Development

The Engineering Laboratory (EL) of the National Institute of Standards and Technology (NIST) has developed a variety of tools that provide some level of economic analysis, many of which have been developed by EL's Applied Economics Office (AEO). AEO-developed tools (both legacy and active described in Table 1-1) range across topic areas and economic metrics: life cycle cost analysis (LCCA), benefit-cost analysis (BCA), sensitivity and uncertainty analysis, risk assessment, and profit maximization / cost minimization.

Table 1-1 AEO-Developed Software

Software Name	Acronym	Topic Area	Analysis
Economic Decision Guide Software	EDGe\$	Community Resilience	BCA
Building for Environmental and Economic Sustainability	BEES	Building Product Sustainability	LCCA
Building Industry Reporting and Design for Sustainability*	BIRDS	Whole Building Sustainability	LCCA
Building Life Cycle Cost	BLCC	Building economics (energy efficiency, sustainability)	LCCA
Bridge Life Cycle Cost	BridgeLCC	Bridge Economics	LCCA
Cost-Effectiveness Tool for Capital Asset Protection*	CET	Building Risk Mitigation	LCCA; Uncertainty analysis; Risk assessment
Sprinkler Use Decisioning*	SpUD	Residential Sprinklers	BCA
Turnout Gear Selector*	ToGS	Firefighter Protection	Risk Assessment
LCCA Tool for Chem/Bio Protection of Buildings*		Facility Chem/Bio protection strategies	LCCA; Rick Assessment
Manufacturing Cost Guide		Manufacturing	Cost Minimization; Profit Maximization; life cycle impact assessment (LCIA)
Monte Carlo Tool		Manufacturing	Uncertainty Analysis
Smart Investment Tool		Manufacturing	LCCA, Uncertainty Analysis
* Legacy Software; no longer supported			

Each tool is targeting government agencies and organizations that fall under NIST's mission. Stakeholders need different levels of details and functionality because of the diverse backgrounds across stakeholders. As a result, AEO has developed these software tools in a "one-off" approach on an as needed basis. Each tool is developed from scratch,

commonly completed by different principal investigators and software developers (some of which are no longer employed at NIST) as different types of software (e.g., Excel Macro-based spreadsheet, desktop application, web application) using different programming languages (e.g., Java, C#, Python) and development tools (e.g., .NET, Django, JavaScript).

As a result, each software has its own unique set of code even though these tools have some key underlying components that are common across tools, including methodology documented in standards. The duplication of efforts has led to ever increasing maintenance costs and/or placing tools into a legacy status with no additional support or development. Additionally, there has been increasing interest in redevelopment and expansion of existing tools (e.g., BLCC) as well as potential new tools targeting other research areas at NIST (e.g., cost-effectiveness of seismic retrofits, prioritization of Border Patrol facility maintenance and improvements). Further software development would continue to stress EL's limited software development resources.

To improve the efficiency of the tools developed by AEO and its stakeholders, an economic service/engine with an associated Application Programming Interface (API) that provides consistent, standardized, transparent, reproducible economic analysis calculations should be developed, maintained, and made publicly available for internal and external researchers, analysts, and software developers. Economic evaluation capabilities should be able to provide life-cycle cost analysis, benefit-cost analysis, profit maximization / cost-minimization, and risk-cost analysis.

The requirements for the tool include that it should be consistent with federal legislation (e.g., EPA 2005, RFA, UMRA, NEPA, PRA), federal regulations (e.g., 10 CFR 436, Circular A-4, Circular A-94), guidance (e.g., Handbook 135, Tri-Services Memorandum of Agreement (MOA)), and executive orders (e.g., EO 12866, 13693, 13123, 13563, and 13579) that define the methodology and necessitate the use of LCCA and/or BCA in decision-making. Additionally, the tool should also be consistent with industry consensus standards (e.g., ASTM) across as many topics as possible, including:

- (1) Buildings (e.g., co-benefits and trade-offs from net-zero energy designs)
- (2) Infrastructure (e.g., life-cycle costs of alternative designs)
- (3) Resilience (e.g., community risk and benefit-cost analysis; seismic and wind building code adoption)
- (4) Manufacturing (e.g., economic assessment of innovation/adoption of new technology)
- (5) Smart Grid (benefit-cost analysis of interoperability)
- (6) Cybersecurity (risk-cost analysis)
- (7) Safety & Security (e.g., risk assessment)
- (8) Fire (e.g., economic impact of fire building code adoption)

Not only could such a tool be used to improve NIST-funded tool development, it could also be leveraged by other federal agencies and national laboratories, state and local

governments, academia, and the private sector to improve their tool development by using a federally developed and validated calculations to provide standard-consistent analysis.¹ A few examples of existing economic analysis tools, both developed and maintained by NIST and other federal agencies, that could leverage such a tool include the following:

- LCCA
 - Department of Energy (DOE) BLCC software (maintained by NIST) for building energy efficiency, water conservation, and renewable energy projects
 - Federal Highway Administration (FHWA) Real Cost (excel-based tool) for highway LCCA
- BCA
 - NIST's EDGe\$ web application for Community Resilience
 - Federal Emergency Management Agency (FEMA) Benefit Cost Toolkit 5.2.1
 - State of Connecticut's Cost/Benefit Analysis Forms
 - FHWA Tool for Operations Benefit-Cost Analysis (TOPS-BC)

1.2 Economic Evaluation Engine (E3)

To address this need, a new software tool has been developed, called the Economic Evaluation Engine (E3). E3 is publicly accessible on a NIST server as well as on GitHub for anyone to use for economic analysis, whether it's through a basic script calling on the E3 API, online interface that connects to the E3 API, or an executable program that is built on E3's capabilities. Additionally, users of E3 could provide expansions to the capabilities by developing the code and submitting it to NIST for review and incorporation.

The potential impacts of successful completion of an all-purpose economic analysis API (E3) would be broad, benefiting every division and office in EL as well as other government agencies (federal and state) that desire economic analysis but do not have the capabilities and/or funding available to develop and continually support similar tools internally. By leveraging E3, NIST researchers will have improved capabilities to evaluate the economic benefits and costs of new technologies and potential changes to or development of new standards and codes, leading to improved design and greater adoption of these standards, codes, and technologies. Additionally, industry could leverage the standards-based API for tool development that could either be used for internal evaluations or provided as a service to other companies, leading to more

¹ Such a tool could be useful for GSA, EPA, Commerce, DHS, NIH, Park Service, VA, DoD, DOE, NASA, State Dept., Smithsonian, USDA, USACE, USAF AFCEC, Army-COE, Navy – NAVFAC, Air National Guard, FEMA, Federal Railroad Admin., Individual States, National Labs, Labor Dept., Dept. of Transportation, US Access Board, US Courts, HHS, Federal Facilities Council, Green Seal, HUD, NIBS, NIST MEP, State Governments, Local Governments.

standardized economic analysis that will better inform decision-making throughout an industry.

E3 will allow AEO to focus its collective resources on maintaining and expanding the API functionality and capabilities to keep it up-to-date and relevant instead of duplicating maintenance efforts across a range of software tools. Tools that leverage the API could be developed by AEO (including transition of existing tools), other EL or NIST researchers, federal and state government agencies, academics, or the private sector (industry groups and individual companies) based on their analysis needs at lower costs because much of the back-end development would already be completed. AEO and ELDST have received positive feedback from leadership across multiple EL research programs. External contacts that have shown an interest in such a tool will also be solicited, including Border Patrol, GSA, DOE FEMP, FHWA, and DHS.

1.2.1 Concept

To put the E3 products into perspective with other related (potential) products, a block diagram has been generated in Figure 1-1. The E3 object library will be built into the API. The E3 API will be maintained on GitHub for 3rd party download, to be used locally with 3rd party scripts, or executable programs developed based on the API as well as be hosted on a NIST server to provide both NIST and 3rd party interfaces web-access. The E3 Interface will be a use case demonstrating the capabilities of the E3 function library and API (specific design to be discussed in Chapter 3). E3 will be designed to allow for expansion through modules that can provide items such as default data, calculation functions based on specific guidance, or more expansive environmental assessments.

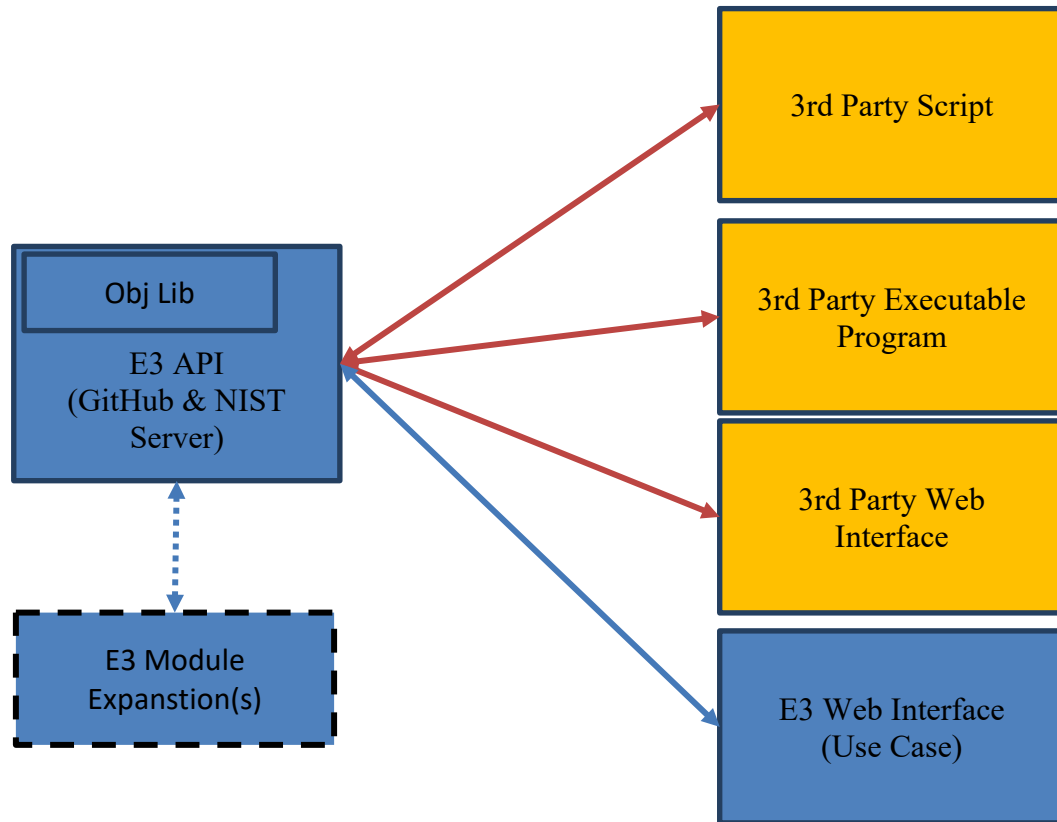


Figure 1-1 Block Diagram of E3 Product Interconnections

At this point in time, there does not exist such a set of products that provide the public with standards-based calculations that are generalized enough to be non-topic specific.

The intended users of the E3 products may include:

- AEO researchers completing economic calculations for internal research
- AEO researchers developing software to complete economic calculations for internal research and/or external parties
- Other NIST researchers wanting to complete economic calculations or develop software to complete economic calculations for internal research and/or external parties
- 3rd party researchers wanting to complete economic calculations or develop software to complete economic calculations

The users will have technical expertise in computer programming and/or software development allowing them to use the products (function library and/or API) without much assistance, primarily met by the support documents provided by AEO/ELDST with minimal technical support from AEO/ELDST staff.

The users will need a conceptual understanding of the economic analysis of interest. Since the users may not have in depth knowledge of economic analysis, the design will need to validate the data submission to ensure reasonableness and provide user's clear feedback on inconsistencies, concerns, or errors (e.g., check object tags for misspelled words that are similar that the user wanted to match).

The general process for a deterministic (single point estimate) analysis within E3 is as follows:

- Receive Request from User
- Authenticate, Validate, and Read the Request
- Calculate the cash flows for each BCN
- Aggregate cash flows for each standard category (i.e., “required” flows)
- Aggregate cash flows for each user-provided, customizable “tag” category (“optional” flows)
- Aggregate BCN flows for each Alternative by standard and tag categories
- Create Measure Summary for each Alternative by comparing to the Baseline Alternative
- Calculate Non-Deterministic Results (Optional)
 - If there are one or more Sensitivity Analyses included in the request, E3 replaces the value in the sensitivity analysis request and replicates the deterministic process. This is replicated for each value in the sensitivity analysis.
 - If there are one or more Uncertainty Analyses included in the request, E3 uses random selection of values that have uncertainty and replicate the deterministic process. This is replicated until the analysis converges or the maximum number of simulations is met. Due to the amount of data, the cash flows generated in an uncertainty analysis are not saved or reportable.
- Generate E3 Output
- Return Output to the User

A user can customize the categorization (i.e., tags), and output objects (e.g., summaries, cash flows by category) to include in the output file. This provides the user the ability to customize the results returned. For example, the user may want to exclude some tags from LCCA results, such as externalities. In this case, the user can call on the cash flow objects and then generate their own results.

Details of the data flow, code, libraries, and classes will be discussed in detail in Chapter 2 and Chapter 3.

1.2.2 Progress/Status

As of the date of this publication, E3 is in beta version. In FY2020, the software requirements specifications document was finalized for designing E3 and examples files

for validating the E3 API were created. In FY2021, the API was developed, internally beta tested, and published on GitHub and a NIST internal web server. Additionally, an example front-end interface has been developed as a use case, which is a LCCA of residential solar photovoltaic systems called Present Value for PhotoVoltaics - [PV]². The initial version of [PV]² includes deterministic analysis. Activities for FY2022 will include external validation as well as incorporating additional analysis features to both E3 and [PV]², including sensitivity and uncertainty analysis, continuous discounting, and non-monetary measures.

1.3 ASTM Standards-Based Methodology

E3 is designed to be consistent with numerous standards from ASTM International, which can define test methods, specifications, classifications, practices, guides, and terminology. The primary focus in the initial development of E3 has been the suite of ASTM Standards developed by Subcommittee E06.81 on Building Economics. However, additional standards have been recently developed that use the same economic methods and apply them to other topic areas, including community resilience and sustainable manufacturing. Although the focus may not be the same and some terminology may be different, the fundamental calculations are consistent across these standards. The standards that are the basis for the E3 design are discussed in the remainder of this chapter.

1.3.1 Building Economics (ASTM E06.81)

ASTM Committee E06 on Performance of Buildings was created to “promote knowledge, stimulation of research, development and maintenance of standards and related documents for performance of buildings, their elements, components, including means and methods of fabrication and assembly; and the description, measurement, prediction, improvement, and management of the overall performance of buildings and building-related facilities” [1]. Subcommittee E06.81 on Building Economics have developed standards for evaluating the potential economic performance of investments in buildings, providing industry standardized approaches for accepting or rejecting an investment, designing and sizing building systems, and establishing priorities. These standards include defining terminology for building economics (E833 [2]), guides for selecting the appropriate economic method (E1185 [3]) and how to summarize economic analysis results (E2204 [4]), and practices that show how to complete analysis using the selected economic methods. The practices will be discussed below by topic area.

1.3.1.1 LCCA and BCA

E06 Standard Practices for Building Economics provide the methods for calculating Life Cycle Cost Analysis (LCCA), Net Savings (NS) and Net Benefits (NB), and supplemental measures of Benefit-Cost Ratio (BCR) and Savings-to-Investment Ratio (SIR), Internal Rate of Return (IRR) and Adjusted Internal Rate of Return (AIRR), and

Payback Period. Each of these measures is summarized below, including their appropriate use in decision making.

E917 is the Standard Practice for Measuring Life-Cycle Costs of Buildings and Building Systems [5]. The LCC method sums all relevant costs associated with an entire building or building system over a specified time period.² Mutually exclusive alternatives for a given functional requirement can be compared based on their LCCs to determine which is the least-cost option over the study period.

The following is the general formula for the LCC present value model:

$$LCC = \sum_{t=0}^N \frac{C_t}{(1+d)^t}$$

Where:

- LCC = Total LCC in present-value dollars of a given alternative
- C_t = Sum of all relevant costs, including initial and future costs, less any positive cash flows, occurring in year t ,
- N = Number of years in the study period
- d = Discount rate used to adjust cash flows to present value

E1074 is the Standard Practice for Measuring Net Benefits and Net Savings for Investments in Buildings and Building Systems [6]. The Net Benefits (NB) method calculates the difference between discounted benefits and discounted costs while the Net Savings (NS) method calculates the difference between LCCs as a measure of the cost-effectiveness of a project, also called the net present value method.

The NB for a project alternative, relative to a designated base case, can be calculated comparing the difference in total benefits and total costs for each alternative:

$$NB = (B_{\text{Alternative}} - B_{\text{BaseCase}}) - (C_{\text{Alternative}} - C_{\text{BaseCase}})$$

Net Benefits can be calculated using individual benefit and cost differences by applying the following general formula:

$$\begin{aligned} NB_{A:BC} &= \sum_{t=0}^N \frac{\Delta B_t}{(1+d)^t} - \sum_{t=0}^N \frac{\Delta C_t}{(1+d)^t} \\ &= \sum_{t=0}^N \frac{\Delta B_t}{(1+d)^t} - \sum_{t=0}^N \frac{\Delta NI_t}{(1+d)^t} - \sum_{t=0}^N \frac{\Delta I_t}{(1+d)^t} = \sum_{t=0}^N \frac{(\Delta B_t - \Delta NI_t - \Delta I_t)}{(1+d)^t} \end{aligned}$$

Where:

- $NB_{A:BC}$ = NB, in PV dollars, of alternative (A), relative to base case (BC)

² E917 allows for providing LCC on an annualized value basis. E3 only reports results in present value terms. The user could use the present value results provided by E3 to provide the annualized values allowed in E917.

ΔB_t	=	Benefit difference in year t associated with the alternative relative to the base case
ΔC_t	=	Cost difference in year t associated with the alternative relative to the base case
ΔNI_t	=	Non-investment cost difference in year t associated with the alternative relative to the base case
ΔI_t	=	Investment cost difference in year t associated with the alternative relative to the base case
t	=	Year of occurrence (where 0 is the base date)
d	=	Discount rate
N	=	Number of years in study period

The NS for a project alternative, relative to a designated base case, can be calculated by simply subtracting the LCC of the alternative from the LCC of the base case:

$$NS = LCC_{BaseCase} - LCC_{Alternative}$$

Net Savings can also be calculated using individual cost differences by applying the following general formula:

$$NS_{A:BC} = \sum_{t=0}^N \frac{S_t}{(1+d)^t} - \sum_{t=0}^N \frac{\Delta I_t}{(1+d)^t}$$

Where:

$NS_{A:BC}$	=	NS, in PV dollars, of alternative (A), relative to base case (BC)
S_t	=	Savings in year t in costs associated with the alternative
ΔI_t	=	Additional investment-related costs in year t associated with the alternative
T	=	Year of occurrence (where 0 is the base date)
D	=	Discount rate
N	=	Number of years in study period

When considering a single project alternative, a project is cost effective if net benefits or net savings are greater than zero. If multiple sizes or design alternatives competing for a given purpose are being considered, the alternative with the greatest net benefits or net savings is the most cost effective.

E964 is the Standard Practice for Measuring Benefit-to-Cost and Savings-to-Investment Ratios for Buildings and Building Systems [7]. The BCR is used when the focus is on (monetizable) benefits relative to project investment costs and compares the combination of benefits and non-investment costs (net benefits excluding investment costs) relative to investment costs. The SIR, a variation of the BCR, is used when the focus is on project savings (i.e., cost reductions) relative to project costs.

The general formula for the BCR rearranges the terms used for NB as a ratio and separates costs into investment costs and non-investment costs:

$$BCR_{A:BC} = \sum_{t=0}^N \frac{(\Delta B_t - \Delta NI_t)}{(1+d)^t} \bigg/ \sum_{t=0}^N \frac{\Delta I_t}{(1+d)^t}$$

Where:

- $BCR_{A:BC}$ = BCR of alternative (A) relative to base case (BC)
- ΔB_t = Benefit difference in year t associated with the alternative relative to the base case
- ΔNI_t = Non-investment cost difference in year t associated with the alternative relative to the base case
- ΔI_t = Investment cost difference in year t associated with the alternative relative to the base case
- t = Year of occurrence (where 0 is the base date)
- d = Discount rate
- N = Number of years in study period

Alternatively, the BCR can be calculated using the NB and net present value of the total investment costs (I):

$$BCR = (NB + I)/I$$

The general formula for the SIR rearranges the two terms for NS as a ratio:

$$SIR_{A:BC} = \sum_{t=0}^N \frac{S_t}{(1+d)^t} \bigg/ \sum_{t=0}^N \frac{\Delta I_t}{(1+d)^t}$$

Where:

- $SIR_{A:BC}$ = SIR of alternative (A) relative to base case (BC)
- S_t = Savings in year t in costs associated with the alternative
- ΔI_t = Additional investment-related costs in year t associated with the alternative
- t = Year of occurrence (where 0 is the base date)
- d = Discount rate
- N = Number of years in study period

A BCR or SIR greater than 1.0 indicates a project is economical because the benefits (benefits minus non-investment costs) or savings (reduction in non-investment costs) are greater than the investment costs. Attention must be taken to correctly define and interpret ratios when using them to choose among multiple alternative designs and sizes of a project or prioritize projects competing for limited funds.

E1057 is the Standard Practice for Measuring Internal Rate of Return and Adjusted Internal Rate of Return for Investments in Buildings and Building Systems [8]. The IRR and AIRR methods measure economic performance as a compound yield on investment. The IRR is the discount rate at which the net present value of future cash flows is equal to the initial investment:

$$NB_{A:BC} = \sum_{t=0}^N \frac{\Delta B_{A,t} - \Delta C_{A,t}}{(1+i^*)^t} = \sum_{t=0}^N \frac{(\Delta B_{A,t} - \Delta NI_{A,t} - \Delta I_{A,t})}{(1+i^*)^t}$$

$$= \sum_{t=0}^N \frac{(\Delta B_t - \Delta NI_t)}{(1+i^*)^t} - \sum_{t=0}^N \frac{\Delta I_t}{(1+i^*)^t} = 0$$

Where:

- $NB_{A:BC}$ = NB of alternative (A) relative to base case (BC)
- ΔB_t = Benefit difference in year t associated with the alternative relative to the base case
- ΔC_t = Cost difference in year t associated with the alternative relative to the base case
- i^* = IRR
- N = Number of years in study period

The same formula can be used for calculating the IRR using NS:

$$NS_{A:BC} = \sum_{t=0}^N \frac{S_t}{(1+i^*)^t} - \sum_{t=0}^N \frac{\Delta I_t}{(1+i^*)^t} = 0$$

- $NS_{A:BC}$ = NS, in PV dollars, of alternative (A), relative to base case (BC)
- S_t = Savings (negative change in non-investment costs) in year t in costs associated with the alternative
- ΔI_t = Additional investment-related costs in year t associated with the alternative
- T = Year of occurrence (where 0 is the base date)
- i^* = IRR
- N = Number of years in study period

The IRR calculation requires the use of a root-finding algorithm due to the nonlinear nature of the IRR formula. Ridders' Method was selected due to its use of bracketing in conjunction with the false-position method. The method attempts to use the false-position method to produce a faster convergence to the root while relying on bracketing if the false position method produces a less optimal estimate. By using the bracketing method as a secondary method, Ridders' method can conditionally guarantee convergence to a root, although at the cost of convergence time (order $\sqrt{2}$ instead of order 2) due to the additional calculations. The condition required for convergence is that the signs of the initial left and right bracketing values must be opposite. This means if there are an even number of roots in the initial bracket (0 to 1) the algorithm will fail. Furthermore, if an even number of roots is present, the algorithm will only converge to one without preference.

The AIRR is the discount rate at which the net present value of future cash flows is equal to the initial investment including earnings on reinvested savings to the end of the study

period. The IRR formula is modified by introducing the reinvestment returns for the NB or NS for each period assuming a constant reinvestment rate (r) compounded for the remainder of the study period:

$$NB_{A:BC} = \sum_{t=0}^N \frac{(\Delta B_t - \Delta NI_t)(1+r)^{N-t}}{(1+i^{**})^N} - \sum_{t=0}^N \frac{\Delta I_t}{(1+r)^t} = 0$$

$$NS_{A:BC} = \sum_{t=0}^N \frac{(S_t)(1+r)^{N-t}}{(1+i^{**})^N} - \sum_{t=0}^N \frac{\Delta I_t}{(1+r)^t} = 0$$

Where:

- i^{**} = AIRR
- r = reinvestment rate
- N = Number of years in study period

If the reinvestment rate is not equal to the discount rate, then the same approach as implemented for calculating the IRR is implemented with the same root-finding algorithm approach.

The most straightforward method of calculating the AIRR requires the calculated BCR for a BCA or SIR for a LCCA for a project (relative to its base case) and assumes the reinvestment rate (r) equals the discount rate. The AIRR can then be computed using the following formulas for BCA and LCCA, respectively:

$$AIRR_{BCA} = (1+r) \cdot \left(1 + \frac{\sum_{t=0}^N \frac{(\Delta B_t - \Delta NI_t)}{(1+r)^N} - \sum_{t=0}^N \frac{\Delta I_t}{(1+r)^t}}{\sum_{t=0}^N \frac{\Delta I_t}{(1+r)^t}} \right)^{\frac{1}{N}} - 1$$

$$= (1+r) \cdot \left(1 + \frac{NB}{I} \right)^{\frac{1}{N}} - 1 = (1+r) \cdot \left(\frac{NB + I}{I} \right)^{\frac{1}{N}} - 1 = (1+r) \cdot (BCR)^{\frac{1}{N}} - 1$$

$$AIRR_{LCCA} = (1+r) \cdot \left(1 + \frac{\sum_{t=0}^N \frac{(S_t)}{(1+r)^N} - \sum_{t=0}^N \frac{\Delta I_t}{(1+r)^t}}{\sum_{t=0}^N \frac{\Delta I_t}{(1+r)^t}} \right)^{\frac{1}{N}} - 1$$

$$= (1+r) \cdot \left(1 + \frac{NS}{I} \right)^{\frac{1}{N}} - 1 = (1+r) \cdot \left(\frac{NS + I}{I} \right)^{\frac{1}{N}} - 1$$

$$= (1+r) \cdot (SIR)^{\frac{1}{N}} - 1$$

Note that in the case where the change in benefits (ΔB_t) for all times t are zero (i.e., LCCA), the benefits only include reductions in non-investment costs (i.e., cost savings)

and the BCR equals the SIR. The reinvestment rate is assumed to be equal to the discount rate. The appropriate value of the reinvestment rate (r) is typically best approximated using the discount rate because a rate higher (or lower) than the discount rate suggests that the discount rate is too low (or too high) because it does not adequately reflect the next best investment opportunity in which you would invest additional cash flows.

The IRR or AIRR is compared against the investor's minimum acceptable rate of return (MARR), and the investment is considered economical if the value is greater than the MARR. IRR and AIRR will be equal if there are only an initial investment cost and the benefit/savings are realized at the end of the study period or if the reinvestment rate (r) equals the IRR. If r is less than the IRR, then the AIRR will be less than the IRR; and if r is greater than the IRR, the AIRR is greater than the IRR. Examples where the IRR and AIRR are both reported as provided in Section 4.5. The AIRR is recommended for most applications in which a return-on-investment measure is desired because it is more reliable than IRR for maximizing NB or NS. Caution is recommended in applying either measure, however, because problems arise under certain conditions.

E1121 is the Standard Practice for Measuring Payback for Investments in Buildings and Building Systems [9]. The payback method determines the number of periods required for the benefits or savings to offset the initial investment costs, and is used for determining whether the investment is recovered within the project's service life or some maximum acceptable payback period (MAPP). The simple payback period (excluding discounting) is the minimum number of years, y , for which

$$\sum_{t=1}^y ((\Delta B_t - \Delta NI_t) - \Delta I_t) \geq \Delta I_0 \text{ or } \sum_{t=1}^y (S_t - \Delta I_t) \geq \Delta I_0$$

Where:

- y = Minimum length of time over which future net cash flows have to be accumulated in order to offset initial investment costs
- S_t = Savings in year t in non-investment costs associated with the alternative
- ΔB_t = Additional benefits in year t associated with a given alternative
- ΔNI_t = Additional non-investment costs in year t associated with a given alternative
- ΔI_0 = Initial investment costs associated with a given alternative
- ΔI_t = Additional investment-related costs in year t , other than initial investment costs

The discounted payback period is the minimum number of years, y , for which

$$\sum_{t=1}^y \frac{(\Delta B_t - \Delta NI_t) - \Delta I_t}{(1 + d)^t} \geq \Delta I_0 \text{ or } \sum_{t=1}^y \frac{S_t - \Delta I_t}{(1 + d)^t} \geq \Delta I_0$$

Where:

y	=	Minimum length of time over which future net cash flows have to be accumulated in order to offset initial investment costs
S_t	=	Savings in year t in operational costs associated with the alternative
ΔB_t	=	Additional benefits in year t associated with a given alternative
ΔNI_t	=	Additional non-investment costs in year t associated with a given alternative
ΔI_0	=	Initial investment costs associated with a given alternative
ΔI_t	=	Additional investment-related costs in year t , other than initial investment costs
d	=	Discount rate

There are numerous reasons why the payback method is widely used, among which are the following. It is easy for a layperson to calculate and understand. It helps to identify projects that will be unusually profitable or unprofitable early in their life. Investors often want to be assured of short paybacks along with high rates of return before investing in assets. However, it is important to note that the decision to use the payback method should be made with care because it is an incomplete measure of economic performance. The primary contribution of the payback method is not for making decisions, but as a supplementary method of economic evaluation. Payback is particularly useful as a screening tool for narrowing the feasible set of projects to which additional economic methods should be applied. The payback method excludes all benefits and costs that occur after the payback period is reached, creating a bias against projects with longer paybacks even if the long-term returns are greater than projects with shorter paybacks. Additionally, payback is not an indication of economic efficiency for a design or size of a project.

1.3.1.2 Uncertainty and Risk

Standards on Building Economics also address how to handle uncertainty, including selecting a technique and developing and evaluating cost risk and risk mitigation strategies.

E1369 is the Standard Guide for Selecting Techniques for Treating Uncertainty and Risk in the Economic Evaluation of Buildings and Building Systems [10]. This guide:

- describes when measuring uncertainty and risk may be helpful
- defines uncertainty, risk exposure, and risk attitude
- presents techniques for measuring uncertainty and risk exposure
- discusses how to select the appropriate technique for a particular problem
- describes how risk exposure can be measured
- describes how risk attitude can be incorporated using utility theory and other approaches

E1946 is the Standard Practice for Measuring Cost Risk of Buildings and Building Systems and Other Constructed Projects [11], which provides a “procedure for measuring cost risk...using the Monte Carlo simulation technique as described in Guide E1369.” Cost risk analysis (CRA) can help answer questions related to uncertainty and risk to the costs of a capital project, such as the likelihood of costs exceeding a target cost level or identifying the aspects of the project that are most impactful on the uncertainty or risk.

E2506 is the Standard Guide for Developing a Cost-Effective Risk Mitigation Plan for New and Existing Constructed Facilities [12]. This guide describes a “generic framework for developing a cost-effective risk mitigation plan for new and existing constructed facilities—buildings, industrial facilities, and other critical infrastructure” and “provides an approach for formulating and evaluating combinations of risk mitigation strategies.” Using this guide in conjunction with the ASTM Practices discussed in Section 1.3.1.1 ensures the inclusion of uncertainty for input parameters in the economic analysis and results reporting.

1.3.1.3 Analytical Hierarchy Process (AHP)

Thus far, the discussion on standards has revolved around monetizable benefits and costs. However, there are often non-monetary benefits and costs that should be considered in the decision-making process, such as location, security, aesthetic environment, image to the public, and environmental externalities. E1765 is the Standard Practice for Applying Analytical Hierarchy Process (AHP) to Multiattribute Decision Analysis of Investments Related to Projects, Products, and Processes, which specifies how to apply the AHP to numerous evaluation criteria with different units of measures to develop a more comprehensive method of evaluation [13]. AHP considers non-monetary criteria for evaluations, potentially qualitative or quantitative, along with common economic criteria to evaluate a project. This practice provides a procedure for calculating and interpreting results from an AHP analysis.

AHP is not explicitly included in E3. However, E3 can develop the different monetary and non-monetary measures necessary as inputs into the AHP approach for post-processing.

1.3.1.4 Applicability of Building Economics Standards

The ASTM Building Economics Standards have been the basis for federal LCCA requirements [14] and guidance for decades, most recently published in the revision to Handbook 135 – Life Cycle Costing Manual for Federal Energy Management Program [15]. Handbook 135 provides examples of evaluating the cost-effectiveness of building projects related to energy efficiency, water conservation, renewable energy, sustainability, and resilience.

Although these standards are written explicitly for evaluating buildings and related engineered systems, the methodologies can be practical for other applications. In fact,

many of these standards are referenced or mentioned in standards covering other topic areas, such as community resilience and manufacturing.

1.3.2 Community Resilience (E06.81)

The first standard guide for cost-effective community resilience strategy decisions, E3130 - Standard Guide for Developing Cost-Effective Community Resilience Strategies, was published in 2018 [16]. E3130 provides guidance for evaluating the economic performance of community resilience-related investment strategies that improve responding, withstanding, and recovering from disruptive events. The guide assists in identifying relevant benefits and costs, including investment costs, future cost savings, and damage loss avoidance, from capital investment into resilience strategies relative to the status-quo. Procedures are provided for defining and satisfying the functions of a project, product, or process. The focus of the cost-effectiveness analysis is on benefit-cost analysis over life-cycle cost analysis because of the important role high-impact, low-probability events have on expected benefits and costs for resilience, including externalities and non-market considerations.

Although the standard takes a slightly different analysis perspective, the analysis methodology is consistent with the Building Economics Standards. The economic evaluations are based on the same standard practices (with the addition of E1699):

- E917 (life-cycle costs)
- E964 (benefit-to-cost and savings-to-investment ratios)
- E1057 (internal rate of return and adjusted internal rate of return)
- E1074 (net benefits and net savings)
- E1121 (payback)
- E1765 (analytical hierarchy process for multi-attribute decision analysis)
- E1699 (value engineering/value analysis) [17]
- E1369 (treatment of uncertainty)
- E2204 (summarizing results of economic evaluations)
- E2506 (cost-effective risk mitigation plan)

E3 provides all the necessary capabilities, including benefit, cost, and non-monetary analysis using the same methodologies defined in E3130.

1.3.3 Sustainable Manufacturing (E60.13)

A new ASTM standard, E3200 - Standard Guide for Investment Analysis in Environmentally Sustainable Manufacturing, covers techniques for evaluating manufacturing investments from the perspective of environmentally sustainable manufacturing by pairing economic methods of investment analysis with environmental aspects of manufacturing [18]. The analysis is focused on trade-offs (or synergies) between (non-monetary) environmental impacts and economic performance.

The economic analysis is based on the same economic methods in the Building Economics Standards for selecting the right economic method and quantifying the economic assessment of an alternative. The metrics included in the standard are net present value, IRR, payback period, and hurdle rate (equivalent to the MARR in E1057). The standard also addresses sensitivity analysis in a consistent manner as the Building Economics Standards in Section 1.3.1.2 on uncertainty and risk.

The environmental assessment requires the quantification of the environmental impact(s) of interest, which are typically in non-monetary units (e.g., carbon dioxide equivalent emissions), for each alternative investment option. The metric can either be total change or percentage change in the unit of measure from the investment relative to the baseline alternative.

The economic assessment and environmental assessment(s) are combined to determine if there is a trade-off and, if so, evaluate the trade-off to determine whether the trade-off in the investment is favorable to the baseline alternative. The appropriate metric depends on the decision type (accept/reject, decision, size, ranking/priority), but each metric is based on relative net present value or IRR and total or percentage changes in the environmental impact value.

There are four metrics for considering a tradeoff where the investment is not environmentally favorable. Maximum impact is the largest acceptable increase in environmental impact, either in percent or total units. Net present value per percent change in environmental impact (NPVP) is the average increase or decrease in income brought about by each percentage point change (or unit change) in environmental impact. Environmental hurdle rate is a value selected by the decision-maker and represents the net present value of income or cost savings needed to compel a 1 % percent increase (or one-unit increase) in environmental impact. The maximum environmental expenditure is the total amount the user is willing to pay to reduce environmental impact after all other financial benefits are considered. Net present value elasticity (NPVE) is the percent increase in net present value per 1 % increase in environmental impact.

There are four metrics for considering a tradeoff where the investment is not financially economical but is environmentally favorable. The maximum environmental expenditure is the total amount the user is willing to pay to reduce environmental impact after all other financial benefits are considered. The maximum environmental expenditure rate (MEER) a value selected by the decision-maker to represent the maximum they are willing to pay to decrease the environmental impact by 1 % (or one unit). The other two metrics are NPVP and NPVE.

The interpretation of NPVP and NPVE is different depending on the trade-off that occurs. For investments that are financially economical but not environmentally favorable, a higher NPVE is considered better. For investments that are environmentally favorable but not financially economical, a lower NPVE is better. The NPVE can be compared between investments, but the NPVE for investments that are financially economical but not

environmentally favorable should not be compared with the NPVE for investments that are environmentally favorable but not financially economical.

Although the standard includes metrics that are not currently calculated in E3, the results from E3 use a consistent methodology and can provide results for both the economic and non-economic (non-monetary) measures if the user submits the necessary information. These results will include total net present value, IRR, hurdle rate/MARR, and payback period as well as non-monetary values (total value and total change in value). These results would be straightforward to post-process into the desired metrics discussed above. An example of economic-environmental trade-off analysis is provided in Section 5.4.

2 Data Flow Through the API

This chapter expands on Section 1.2.1 by providing details on the libraries and classes in E3 as well as a walk-through of the workflow logic (pseudocode/actual code). Figure 2-2 shows the information workflow of E3, which can be categorized into user defined classes, main and validation classes, libraries, and API-created classes. Chapter 2 and Chapter 3 are meant to be introductions to the code for those interested in understanding and possibly modifying the API files. This section will serve as a guide to the basics of the flow of data through the calculations. Libraries and Classes necessary to run the calculation will be briefly defined and the diagram in Figure 2-2 will be explained, providing a basic understanding of the steps involved. Section 2.3 provides specifics on the libraries themselves, including their dependencies, their functions, and their classes. These are meant to work in tandem with the publicly available code and variable dictionary available in the E3 GitHub repository, or Section 3, to allow a user to easily follow the process from start to finish.

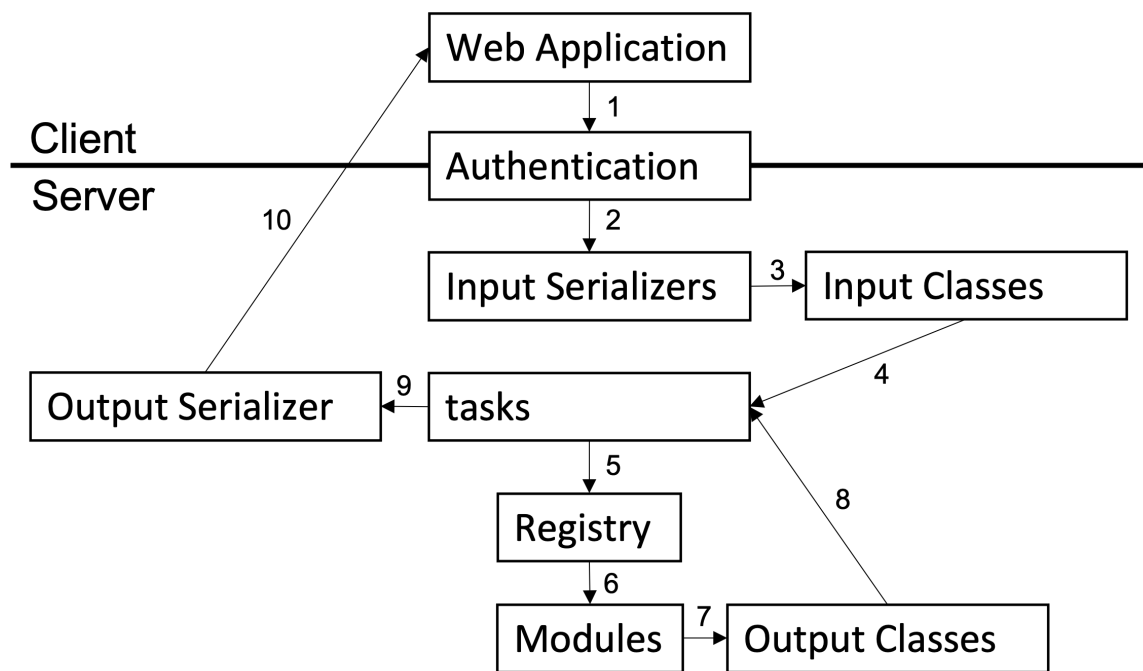


Figure 2-2 E3 Information Workflow

Before fully exploring the flow through Figure 2-2, the individual libraries and classes involved need to be defined. Section 2.2 and Section 2.3 provide the required background. Section 2.4 provides the high-level walkthrough of data flow through the calculations. Non-calculation libraries are omitted from the current discussion for brevity.

2.1 Directory Structure for Files Mentioned in Figure 2-2

A brief overview of the key paths within the directory structure for files specific to calculations is presented here. See the code on GitHub for the full structure.

The *tasks* and *registry* files can be found in E3/e3_django/API

Input classes and the Output class can be found in E3/e3_django/API/objects

Input serializers and the Output serializer can be found in E3/e3_django/API/serializers

All modules can be found in E3/e3_django/compute (objects and serializers directories are not modules)

Output classes, excluding the Output class itself, can be found in E3/e3_django/compute/objects

Output serializers, excluding the Output serializer itself, can be found in E3/e3_django/compute/serializers

2.2 Input Classes (User-Defined Classes)

Input Classes are the objects (located in *objects* directories) provided by the user in the E3 input file converted from JSON to Python format via their respective serializers (located in *serializers* directories). The user must provide an analysis-level set of information (Analysis Class), details on each benefit, cost, and non-monetary value for each “project” alternative including the baseline alternative (Benefit/Cost/Non-monetary (BCN) Class), and any required sensitivity (Sensitivity Class) and/or uncertainty (Uncertainty Class) analysis the user desires to be completed. Each is described below. It is important to note that the server side will handle these objects as both object classes and serializers. Serializers are a class of functions that allow data to be quickly converted from a structured data format non-readable by a programming language to one that is readable and vice versa. For Input Classes, the serializer serves to convert the incoming JSON input into Python objects while simultaneously validating them. The heading name includes both the object class itself, with the associated serializer in parenthesis.

2.2.1 Analysis (AnalysisSerializer)

The analysis class serves to store all inputs that are consistent across all aspects of the analysis. These include values such as the discount rate, study period, and whether real or nominal discounting is used. It also contains descriptive aspects of the analysis that, at present, are not used in the analysis itself. Information like the physical address of the project, what the project is and what type of project is being analyzed (Building, Energy, etc.). Input validation is also performed within the class. The type of output desired is also defined in the Analysis class. More detail on the Analysis class can be found in Section 3.2.1.1.

2.2.2 Alternative (AlternativeSerializer)

The Alternative class stores information related to a specific alternative. It doesn't serve a role in the calculations themselves but is instead meant to allow users to organize and quickly pull information for a specific alternative. Input validation is also performed within the class. More detail on the Alternative class can be found in Section 3.2.1.2.

2.2.3 Benefit / Cost / Non-monetary (BcnSerializer)

Benefit/Cost/Non-monetary (BCN) objects store all information required to determine the cash flow (and quantity flow) for a specific cost, benefit, or non-monetary account item. It also relates how a BCN links to alternatives. Input validation is performed within the class. Some calculations are performed as well, including the generation of BCN level cash flows, quantity flows, discounting, and residual value. More detail on the BCN class can be found in Section 3.2.1.3.

2.2.4 Sensitivity (SensitivitySerializer)

Sensitivity Objects (Input) store information related to any sensitivity analysis a user may wish to run. Sensitivity is the standard "one at a time" analysis and only allows change in one direction. A plus-minus change therefore requires two objects to be created. The object keeps information on the BCN object to be altered as well as the specific variable being changed, the type of change (gross or percentage), and the value of the change. More detail on the Sensitivity class can be found in Section 3.2.1.4.

2.2.5 Input (InputSerializer)

The input class takes the user defined inputs and collects them in a single object. The corresponding serializer then converts them into python objects for use in API calculations. More detail on the Input class can be found in Section 3.2.1.1.

2.3 Libraries

Apart from class/serializer libraries, there are multiple libraries dedicated to calculations required to build and populate the output objects. This section will provide a brief overview of the API libraries used to complete calculations as well as those classes whose attributes are defined by the API. These can be broadly categorized into those required for calculations (Modules) and those required to make the API able to receive, process and return requests (Support Libraries).

2.3.1 Support Libraries

Support libraries don't run any calculations required to arrive at output but facilitate them either by organizing code for ease of debugging and maintenance, or providing functionality required to allow the API to receive, run, and return requests. The libraries in this section are only those required to pass data for calculations purposes.

There are numerous files that are strictly related to server functionality, app construction, and authentication. These are not covered directly here. Instead, an example of how to set up E3 to run locally (Section 4.3) is presented. Any key libraries that need to be understood to do so are explained there. The detail here is intended to allow data flow to be understood.

2.3.1.1 tasks

The library responsible for organizing library calls is called *tasks*. This is identical in purpose to a *Main* library. It receives the input objects, passes them through all required libraries and modules and lastly calls on output creation and formatting before sending to the client. More information on tasks can be found in Section 3.2.4.1.

2.3.1.2 registry

The registry defines the requirements to add and run a module in E3. Modules themselves are collections of code that perform a group of calculations required to arrive at a specific output. These calculations are defined in the respective *app* library within each module directory. The Modules used in the current version of E3 are defined in Section 2.3.2.

2.3.2 Modules

E3 uses modules in lieu of a rigid directory structure to allow functionality to be added with relative ease. Modules themselves include multiple files that define their behavior, however for the purposes of the current E3 version only the *apps* file requires the non-default code. The *app* file contains the calculations specific to the module being developed as well as the definitions for how the Module Graph defined in the registry calls these calculations and what output and dependencies it expects for them. An example of how to add a module to expand functionality is under development using NIST's EDGe\$ tool as a basis [19].

2.3.2.1 cashflow

The cashflow module handles all calculations required to arrive at cash flows for individual BCNs, required flows, and optional (tag) flows by time step in conjunction with the classes defined in the CashFlow library (CashFlow class and OptionalCashFlow class). The cashflow module mostly serves to organize the calls to create the appropriate objects while the bulk of the calculations are done in CashFlow library. These calculations are always done and can be returned as an individual report or used as input in calculating other objects. Section 3.2.3.1 contains detailed information on the cashflow module.

2.3.2.2 measures

The measures module is used to calculate economic measures in conjunction with the AlternativeSummary class. These include BCR, SIR, and itemized NPVs of benefits and costs. IRR is also calculated here if the appropriate object is selected by the user to report.

Measures mostly serves to organize the calls to create the appropriate objects while the bulk of the calculations are done in AlternativeSummary. Section 3.2.3.2 contains detailed information on the measures module.

2.3.2.3 optional

The optional module contains all calculations required to arrive at the cash flows for tagged BCN objects by timestep. It also stores these flows in a separate class for ease of access. Section 3.2.3.3 contains detailed information on the optional module.

2.3.2.4 required

The optional module contains all calculations required to arrive at the cash flow for all BCN objects by timestep broken down by type, subtype, and their combinations. These flows are stored in both their discounted and non-discounted forms. It also stores these flows in a separate class for ease of access. Section 3.2.3.4 contains detailed information on the required module.

2.3.2.5 sensitivity

The sensitivity module contains the calculations required to complete sensitivity analysis in conjunction with the Sensitivity class. The process also involves rerunning the cashflow calculations for any BCNs altered by the sensitivity input. Section 3.2.3.5 contains detailed information on the sensitivity module.

2.3.2.6 Module Workflow

The cashflow module is required to run for any analysis as it generates the BCN flows used in all calculations. Once the cash flows are created, they are made available to all other modules. The required and optional modules work in conjunction with their respective classes in the CashFlow library to generate the aggregated cash flows for all BCNs by timestep. The measure module, with the AlternativeSummary class, then uses the results of the cashflow module to determine final measure and sum pertinent cash flows to their aggregates and stores them as AlternativeSummary objects.

Sensitivity can either repeat the entire process above, or it can work in a more targeted fashion depending on whether the discount rate is the variable being altered in the analysis. Since the discount rate impacts all BCNs any change to it requires a full rerun of the analysis. If a single BCN is changes however, it can be singled out and only its cash flows will be rerun. Regardless of the way cash flows are generated the process from generating cash flows to calculating measures is redone, with the only difference being the output is now stored in Sensitivity class objects.

2.3.3 Output Classes

Output Classes are used to store the results of calculations as python objects (located in *objects* directories). After all calculations are complete the output classes are converted to

a single output object and, via their individual serializers (located in *serializers* directories), converted into a JSON format to be sent back to the user. This section includes the output classes currently available in E3 and their respective serializers.

2.3.3.1 CashFlow (CashFlowSerializer)

The CashFlow library contains the CashFlow and OptionalCashFlow classes. These classes calculate and store the cash flows for the aggregated BCNs and tagged BCNs, respectively. Section 3.2.2.2 contains detailed information on the CashFlow library.

2.3.3.2 AlternativeSummary (AlternativeSummarySerializer)

The AlternativeSummary library is used to calculate and store the economic measures for a single alternative. It also defines the AlternativeSummary Class. More information on the AlternativeSummary library and the AlternativeSummary Class can be found in Section 3.2.2.1.

2.3.3.3 SensitivitySummary (SensitivitySummarySerializer)

The SensitivitySummary library is only used to store the output for sensitivity calculations in the SensitivitySummary class. The calculations to arrive at the measures defined in it are calculated in the AlternativeSummary library. Section 3.2.2.3 contains detailed information on the SensitivitySummary library.

2.3.3.4 Output (OutputSerializer)

The Output library contains the Output class, which stores all individual output objects (required flows, optional flows, sensitivity summary, and measures) in a singular Output class for ease of generating final output to the user. Section 3.2.2.4 contains detailed information on the Output library.

2.4 Data Flow through the API

With the libraries defined, the flow through the API in Figure 2-2 can be described. Only the calculations section is represented here to avoid the flow diagram from becoming too cluttered.

Note that the flow here does not reference all serializers in the code. Many of them are currently not used but are meant to be available should intermediate output be desired. By having the serializers already constructed, converting any step into a JSON string for output becomes much simpler. The list below intentionally lacks fine details. It is meant as a guide to help follow the code and variable dictionary available in the E3 GitHub repository, in conjunction with the more detailed information in 3.2.

1. User generates an input JSON string and sends a request to the API for analysis
2. API authenticates the request and passes input through to the input serializers

-
3. The input serializers convert user input (in JSON format) to input classes (in python format)
 4. The input objects created by their respective classes are passed to the tasks library to begin calculations
 5. Tasks calls to registry to pull the module graph information to start running modules
 6. Registry begins running modules based on user selected objects to report
 7. Modules construct output classes
 8. Output classes are returned to the tasks library
 9. The tasks library uses output serializers to convert the python classes to json format
 10. Tasks returns the JSON formatted output to the user

3 Code, Library, and Class Details

3.1 Development Tools

The current API was coded in Python using a Django REST Framework and deployed in a Docker container on an Amazon Web Services (AWS) instance. See Section 3.3 for future changes to the development tools for the forthcoming update to E3.

3.1.1 Programming – Python, numPy, and pytest

Python is an open source object-oriented, interpreted, and interactive programming language. Python combines power with clear syntax, and has modules, classes, exceptions, high-level dynamic data types, and dynamic typing. There are interfaces to many system calls and libraries, as well as to various windowing systems. New built-in modules are easily written in C or C++ (or other languages, depending on the chosen implementation). Python is also usable as an extension language for applications written in other languages that need easy-to-use scripting or automation interfaces. Python can be used for web, graphical user interface (GUI), and software development, system administration, and scientific and numeric analysis. NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more. Pytest is a mature full-featured Python testing tool. With pytest, common tasks require less code and advanced tasks can be achieved through a variety of time-saving commands and plugins. It will even run your existing tests out of the box including those written with Python's unittest module.

3.1.2 Framework – Django, Django REST Framework, and Django.test

Django is an open source, high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, allowing for focusing on writing the app without needing to reinvent the wheel. Django includes dozens of extras you can use to handle common web development tasks. Django takes care of user authentication, content administration, site maps, Resource Description Framework (RDF) Site Summary or Really Simple Syndication (RSS) feeds, and many more tasks. Django helps developers avoid many common security mistakes, such as Structured Query Language (SQL) injection, cross-site scripting, cross-site request forgery and clickjacking. Its user authentication system provides a secure way to manage user accounts and passwords. Django's quick and flexible scale can meet heavy traffic demands. Django can be used build content management systems to social networks to scientific computing platforms.

Django REST (representational state transfer) framework is a powerful and flexible toolkit for building Web APIs. REST framework has several benefits, including the web browsable API, authentication policies including packages for OAuth1a and OAuth2, serialization that supports both object-relational mapping (ORM) and non-ORM data sources, completely customizable, and extensive documentation, and excellent community support.

The Django test client (`django.test`) is a Python class that acts as a dummy web browser, allowing testing of views and interact with Django-powered applications programmatically. Some of the things test client can be used for include:

- Simulate GET and POST requests on a URL and observe the response.
- See the chain of redirects (if any) and check the URL and status code at each step.
- Test that a given request is rendered by a given Django template, with a template context that contains certain values.
- Use Django's test client to establish that the correct template is being rendered and that the template is passed the correct context data.
- Use in-browser frameworks like Selenium to test rendered HTML and the behavior of Web pages, namely JavaScript functionality.

3.1.3 Deployment – PostgreSQL, Docker, AWS

PostgreSQL is a powerful, open source, highly extensible object-relational database system with a strong reputation for reliability, data integrity, robust feature set, extensibility, and the dedication of the open source community behind the software to consistently deliver performant and innovative solutions. PostgreSQL runs on all major operating systems, has been ACID-compliant since 2001, and has powerful add-ons such as the popular PostGIS geospatial database extender.

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production. Docker provides the ability to package and run an application in a loosely isolated environment called a container. The isolation and security allow you to run many containers simultaneously on a given host. Containers are lightweight and contain everything needed to run the application, so you do not need to rely on what is currently installed on the host. You can easily share containers while you work and be sure that everyone you share with gets the same container that works in the same way.

AWS offers information technology infrastructure services to businesses in the form of web services (i.e., cloud computing). The E3 Docker Container is hosted on an AWS instance.

3.1.4 Input/Output Format

The input for the E3 API is expected as a JavaScript Object Notation (JSON) string. A JSON string organizes input into a collection of key names, and values defined for those key names. Utilizing JSON strings allows for a simple means of standardizing input/output objects creation and ensures that data is received in a usable format. Below is a simple example input string and output string. For additional, more detailed examples see Section 4.5.

3.2 E3 Files

This section provides details on the files defining the E3 API. The goal of this section is not to cover every possible conditional or track every variable in the code, but instead is meant to provide a guide to the libraries, functions, and classes available in the code base. Coupling Sections 2 and 3 with the code itself should provide the user with sufficient information to work with the calculation code effectively. Section 4.2 and Section 4.3 provide basic guidance on sending requests to the E3 API server and running the E3 API on a local machine, respectively.

All sections contain the following: a description of the library's purpose, all libraries/classes calling on the library, all libraries and classes called by the library, and all functions in the library (a function is defined herein as any code following a *def* statement that does not initiate a class or a method). All functions are further broken down into their purpose, inputs, and outputs. If a library contains a class, that class will be broken down into its purpose, attributes, and any methods (definition is the same as that for a function except it is within a class) it contains (with the same breakdown as mentioned for functions). While all classes have an associated serializer, they will not be discussed. As they exist to convert data between formats (JSON to python and vice versa), they are self-explanatory. In all cases the serializers have the same attributes as the classes they are converting. Required libraries only refer to those that are initiated via an *import* call.

3.2.1 Input Classes (User-Input Classes)

3.2.1.1 Analysis (AnalysisSerializer)

Purpose: The analysis class serves to store all inputs that are consistent across all aspects of the analysis.

Required Libraries: None

3.2.1.1.1 Functions

calculate_inflation_rate

Purpose: Calculates inflation rate from the nominal and real discount rates

Input: discount_rate_nom, discount_rate_real

Output: inflation rate

calculate_discount_rate_nominal

Purpose: Calculates the nominal discount rate from the real discount rate and the inflation rate

Input: inflation_rate, discount_rate_real

Output: nominal discount rate

calculate_discount_rate_real

Purpose: Calculates the real discount rate from the nominal discount rate and the inflation rate

Input: discount_rate_nominal, inflation_rate

Output: real discount rate

3.2.1.1.2 Classes

Analysis

Purpose: Stores values defined by user for an Analysis object

`__init__` inputs: self, analysisType, objToReport, studyPeriod, baseDate, timestepVal, timestepComp, Marr, noAlt, baseAlt, reinvestRate, projectType, serviceDate, outputRealBool, interestRate, dRateReal, dRateNom=None, inflationRate, incomeRateFed, incomeRateOther=None, location=None

Attributes: Table 3-2

Table 3-2 Analysis Variable Details

Variable	Var Name	Var Type	Format/Values	Req
Analysis Type	analysisType	String	{LCCA, BCA, Cost-Loss, Profit Maximization, Other}	X
Project Type	projectType	String		
ObjectsToReport	objToReport	List of Strings	[FlowSummary, MeasureSummary, OptionalSummary, SensitivitySummary, IRRSummary*, UncertaintySummary**]	X
Study Period	studyPeriod	Int	Integer	X
Base Date	baseDate	Date	Year-Month-Day	X
Service Date	serviceDate	Date	Year-Month-Day	
TimestepValue	timestepVal	String	Year, Quarter, Month, Day	X
TimestepCompound	timestepComp	String	{EndOfYear, MidYear, Continuous}	X
Output in Nominal or Real Dollars	outputRealBool	Boolean	{Nominal, Real} or {0,1}	X
Interest Rate (Nominal)	interestRate	Float	in decimals	
Discount Rate (Real or Nominal)	dRateReal	Float	in decimals	X
	dRateNom	Float		
Inflation Rate	inflationRate	Float	in decimals	
MARR (Real or Nominal)	Marr	Float	in decimals	X
Reinvestment Return Rate (Real or Nominal)	reinvestRate	Float	in decimals	X
Income Tax Rate	incomeRateFed	Float	Federal	
	incomeRateOther	Float	State/Local	
Location	location	List of Strings	Country, Region, Division, State, County, City, ZIP, address	
Number of Alternatives	noAlt	Int	whole digits	X
Baseline Alternative Number	baseAlt	Int	whole digits	X
* IRRSummary is its own report because it requires convergence calculations.				
** UncertaintySummary is not currently available and will be available in the next version of E3.				

Methods: None

3.2.1.2 Alternative (AlternativeSerializer)

Purpose: The Alternative class stores information related to a specific alternative.

Required libraries: None

3.2.1.2.1 Functions

None

3.2.1.2.2 Classes

Alternative

Purpose: The Alternative class stores information related to a specific alternative.

`__init__` inputs: self, altID, altName, altBCNList, baselineBool

Attributes: Table 3-3

Table 3-3 Alternative Variable Details

Variable	Var Name	Var Type	Format/Values	Req
Alternative ID	altID	int	N/A	X
Alternative Name	altname	String	N/A	
Alternative BCN List	altBCNList	List of ints	N/A	X
Baseline Boolean	baselinBool	Boolean	true/false	X

Methods: None

3.2.1.3 BCN (BcnSerializer)

Purpose: Benefit/Cost/Non-monetary (BCN) objects store all information required to determine the cash flow for a specific cost, benefit, or non-monetary account item.

Required libraries: math, typing, variables

3.2.1.3.1 Functions

create_list

Purpose: initializes a list of set size to store a cash flow

Input: Size

Output: a list of length size plus one

3.2.1.3.2 Classes

Bcn

Purpose: stores all values related to a Bcn and makes calls to calculate a Bcn's cash flows

`__init__` inputs: self, studyPeriod, **kwargs

Attributes: Table 3-4

Table 3-4 BCN Variable Details

Variable	VarName	VarType	Type
AlternativeID	altID	List of Int	Whole digit array; object can apply to more than 1 alternative
BCN ID	bcnID	Int	Whole Digit
Type	bcnType	String	{Benefit, Cost, NonMonetary} or {0,1,2}; revenue treated as negative cost
SubType	bcnSubType	String	{Direct, Indirect, Externality} or {0,1,2}
Name	bcnName	String	String
Tag (Custom Category)	bcnTag	String	String
DateofOccurrence	initialOcc	Int	Whole digit
NominalorReal	bcnRealBool	Boolean	{true, false} or {0,1}; true means real value
Investment	bcnInvestBool	Boolean	{true, false} or {0,1}; true means investment
InvestmentServiceLife	bcnLife	Int	Digit
ResidualValueBoolean	rvBool	Bool	{true, false} or {0,1}; true means to calculate a residual value associated with this object.
ResidualValueOnly	rvOnly	Bool	{true, false} or {0,1}; true means the object is a residual value object and is calculated as a negative investment cost; false means that the object should be used to generate both an investment cost value and a residual value at the end of study period using Linear Depreciation
Recurrence	recurBool	Bool	{true, false} or {0,1}
RecurrenceInterval	recurInterval	Int	Digit
RecurrenceVariabilityRateType	recurVarRate	String	{Percent Delta Timestep X-1}
RecurrenceVariabilityRateValues	recurVarValue	List of Floats	List or Float (if constant)
RecurrenceEndDate	recurEndDate	Int	Whole digit
ValuePerQ	valuePerQ	Float	Float
Quantity	quant	Float	Float
QuantityVariabilityRateType	quantVarRate	String	{Fraction Delta Timestep X-1}
QuantityVariabilityRateValues	quantVarValue	List of Floats	List or Float
Quantity Unit	quantUnit	String	String

Methods:

__repr__

Purpose: Returns the BCN's id

Input: self

Output: BCN ID

3.2.1.4 Sensitivity (SensitivitySerializer)

Purpose: The Sensitivity library defines the Sensitivity Class used to store user-input related to sensitivity, as well as functions to update BCN or analysis variables for sensitivity calculations.

Required libraries: logging, copy, math, decimal, rest_framework.exceptions

3.2.1.4.1 Functions

validate_time

Purpose: Checks that alterations made to time related variables do not result in invalid values, for instance an initial occurrence after the end date for a recurring BCN.

Input: varName, value, bcnObj, study_period

Output: None if valid, raises exception if invalid

3.2.1.4.2 Classes

Sensitivity

Purpose: Stores user input related to sensitivity.

__init__ inputs: self, globalVarBool, altID, bcnID, bcnObj, varName, diffType, diffValue

Attributes: Table 3-5

Table 3-5 Sensitivity Object Details

Variable	Var Name	Var Type	Format/ Values	Req
Boolean for Discount Rate	globalVarBool	Boolean	true/false or None. None corresponds to False	
List of impacted alternatives by ID	altID	List of Ints	Required if globalVarBool is False. Only used for reporting purposes	X
BCN being altered by ID	bcnID	Int	Required if globalVarBool is False.	X
BCN name	bcnObj	String	Not required input, the API will fill in based on altID	
BCN variable to be altered	varName	String	Valid entries are: "discountRate", "initialOcc", "bcnLife", "recurValue", "recurEndDate", "valuePerQ", "quant", "quantValue"	X
Nature of the alteration to the BCN value	diffType	String	Valid entries are: "Gross", "Percent"	X
Amount the BCN value is changed	diffValue	Float	Can be negative or positive	X

Methods:

calculateOutput

Purpose: Returns the altered BCN value for use in the sensitivity calculation

Input: self, base_input, analysis (optional)

Output: new_bcn if globalVarBool is False, discount_rate_new if globalVarBool is True

3.2.1.5 Input (InputSerializer)

Purpose: Stores all inputs generated by the user in a single object

Required libraries: typing, API.objects

3.2.1.5.1 Functions

None

3.2.1.5.2 Classes

Input

Purpose: Stores all inputs generate by the user in a single object

`__init__` input: self, analysisObject: Analysis, alternativeObjects: List[Alternative], bcnObjects, sensitivityObjects=Sensitivity, scenarioObject

Attributes: Table 3-6

Table 3-6 Input Object Details

Variable	Var Name	Var Type	Format/Values	Req
Analysis Object	analysisObject	Object	N/A	X
Alternative Objects	alternativeObjects	List of Objects	N/A	X
BCN Objects	bcnObjects	List of Objects	N/A	X
Sensitivity Object	sensitivityObject	Object	N/A	
Scenario Objects	scenarioObjects	List of Objects	N/A	

Methods: None

3.2.2 API Defined Classes

3.2.2.1 AlternativeSummary (AlternativeSummarySerializer)

Purpose: Calculates measures and stores output for a single alternative

Required libraries: typing, numpy, operator, math, API.variables, compute.objects, decimal

3.2.2.1.1 Functions

net_benefits

Purpose: Takes benefits and costs from the baseline alternative and another alternative to calculate the net benefits of a single alternative

Input: benefits, costs, benefits_base, costs_base

Output: net benefits

net_savings

Purpose: Takes costs from the baseline alternative and another alternative to calculate the net savings for a single alternative

Input: costs, costs_baseline

Output: net savings

bcr

Purpose: Takes benefits and costs from both the baseline alternative and another alternative to calculate the BCR

Input: benefits, costs_inv, costs_non_inv, costs_base, costs_non_inv_base

Output: benefit-cost ratio

sir

Purpose: Takes investment and non-investment costs from both the baseline alternative and another alternative to calculate the savings to investment ratio.

Input: costs_inv, costs_non_inv, costs_inv_base, costs_non_inv_base

Output: savings-investment ratio

check_fraction

Purpose: Checks the numerator and denominator of a fraction to determine if the value should be Infinity, NAN, or the actual value

Input: numerator, denominator

Output: if denominator ≤ 0 and numerator > 0 , Infinity

If denominator ≤ 0 and numerator ≤ 0 , NAN

Else numerator/denominator

discArray

Purpose: Creates and returns an array of discounting multipliers for use in the IRR calculation

Input: size, compounding, rate

Output: array

rateFunction

Purpose: Calculates the value of the function being used by riddersMethod. Kept separate in the event additional functions are added

Input: size, values, compounding, rate

Output: functional value for use in riddersMethod

riddersMethod

Purpose: Implementation of Ridders' method for root finding to calculate the IRR for a given alternative

Input: values, compounding

Output: IRR if root is found, None if no root is found

irrMeas

Purpose: Calculates non-discounted flows for IRR calculation and makes call to riddersMethod to obtain value

Input: costs_nondisc, bens_nondisc, costs_base_nondisc, bens_base_nondisc, comp_type

Output: value obtained from riddersMethod

termValArray

Purpose: Generates an array of multipliers used to arrive at the terminal value of a cash flow. Used by air

Input: size, compounding, rate

Output: array

airr

Purpose: Calculates the adjusted rate of return

Input: net_besns, tot_costs_inv, tot_costs_inv_base, reinvest_rate, d_rate, study_period, flow, baseline, compounding

Output: airr

payback_period

Purpose: Takes the benefits and costs from an alternative and calculates the payback period

Input benefits, baseline_benefits, costs, baseline_costs

Output: payback period

ns_per_q

Purpose: Takes net savings and the change in quantity from the baseline alternative to calculate the net savings per change in quantity

Input: savings, delta_q

Output: net savings per change in quantity

ns_per_pct_q

Purpose: Takes net savings, the total quantity from the baseline, and the change in quantity from the baseline alternative to calculate the net savings per percent change in quantity

Input: savings, delta_q, total_q_base

Output: net savings per percent change in quantity

ns_elasticity

Purpose: Takes net savings and total costs from an alternative, the total quantity from the baseline, and the change in quantity from the baseline alternative to calculate the net savings per percent change in quantity

Input: savings, total_costs, delta_q, total_q_base

Output: elasticity

calculate_quant_sum

Purpose: Calculates the sum of a given set of optional non-monetary flows

Input: optionals³

Output: list of total tag flows

calculate_quant_units

Purpose: Collects the units for the quantities totaled in calculate_quant_sum

Input: optionals

Output: list of units

calculate_delta_quant

Purpose: Takes optional flows and calculates the change in quantities from the baseline alternative

Input: optionals, baseline

Output: list of changes in quantity

calculate_ns_perc_quant

³ optionals refers to optional flow objects or optional flows

Purpose: Calculates net savings percent quantities for the given optional flow objects using the given baseline optional flow objects and this alternative's net savings.

Input: savings, optionals, baseline

Output: list of net savings per percent change in quantity

calculate_ns_delta_quant

Purpose: Calculates ns delta quantities for the given optional flow objects

Input: savings, delta_q, optionals

Output: list of net savings per change in quantity

calculate_ns_elasticity

Purpose: Calculates net savings elasticity quantities for the given optional flow objects using the given baseline optional flows.

Input: savings, total_costs, optionals, baseline

Output: list of net savings elasticities

3.2.2.1.2 Classes

AlternativeSummary

Purpose: Performs calculations to determine measures and stores them for a single alternative

`__init__` inputs: self, alt_id, reinvest_rate, study_period, marr, d_rate, flow:

RequiredCashFlow, optionals: List[OptionalCashFlow], timestep_comp, baseline:

"AlternativeSummary", include_irr: bool

Attributes: Table 3-7

Table 3-7 Alternative Summary Variable Details

Variable	Var Name	Var Type	Format/Values	Req
Alternative ID	altID	Int	N/A	X
Total Benefits Flow	totalBenefits	Float	N/A	X
Total Costs Flow	totalCosts	Float	N/A	X
Total Investment Costs	totalCostsInv	Float	N/A	X
Total Non-Investment Costs	totalCostsNonInv	Float	N/A	X
Total Subtype Flows	totalSubtypeFlows	List of Floats	N/A	X
Net Benefits	netBenefits	Float	N/A	
Net Savings	netSavings	Float	N/A	
Savings to Investment Ratio	SIR	Float	N/A	
Internal Rate of Return	IRR	Float	N/A	
Adjusted Internal Rate of Return	AIRR	Float	N/A	
Simple Payback Period	SPP	Float	N/A	
Discounted Payback Period	DPP	Float	N/A	
Benefit cost Ratio	BCR	Float	N/A	
Quantity Sum	quantSum	List of Floats	N/A	
Quantity Units	quantUnits	List of Floats	N/A	
MARR	MARR	Float	N/A	
Change in Quantity	deltaQuant	List of Floats	N/A	
Net savings per change in quantity	nsPercQuant	List of Floats	N/A	
Net savings per change in percent quantity	nsDeltaQuant	List of Floats	N/A	
Net savings elasticity	nsElasticityQuant	List of Floats	N/A	

Methods:

__str__

Purpose: Allows object variables to be printed using a print statement. Only used for debugging purposes

Input: self

Output: list of values for an instance of the object class

3.2.2.2 CashFlow (CashFlowSerializer)

Purpose: The CashFlow library defines the CashFlow class and the RequiredCashFlow and the OptionalCashFlow subclasses

Required libraries: operator, API.objects, API.variables

3.2.2.2.1 Functions

elementwise_add

Purpose: Adds two lists of equal length element by element

Input: x, y

Output: elementwise addition of x and y

bcn_type

Purpose: Decorator to denote when a function should only run for a certain cost type

Input: bcn_type_parameter

Return: decorator for the given bcn_type_parameter

3.2.2.2.2 Classes

CashFlow

Purpose: Defines the base cash flow object. Full functionality is added through subclasses

Attributes: None

Methods:

add

Purpose: provides functionality to add BCNs to a cashflow

Input: self

Output: None

RequiredCashFlow

Purpose: Sums cash flows and stores the results for a single alternative

`__init__` input: self, alt: int, study_period: int

Attributes: Table 3-8

Table 3-8 Required Cash Flow Variable Details

Variable	Var Name	Var Type	Format /Values	Req
Alternative ID	altID	Int	N/A	X
Total Non-Discounted Costs	totCostNonDisc	List of floats	N/A	X
Total Discounted Costs	totCostDisc	List of floats	N/A	X
Total Non-Discounted Benefits	totBenefitsNonDisc	List of floats	N/A	X
Total Discounted Benefits	totBenefitsDisc	List of floats	N/A	X
Total Non-Discounted Investment Costs	totCostsNonDiscInv	List of floats	N/A	X
Total Discounted Investment Costs	totCostsDiscInv	List of floats	N/A	X
Total Non-Discounted Investment Benefits	totBenefitsNonDiscInv	List of floats	N/A	X
Total Discounted Investment Benefits	totBenefitsDiscInv	List of floats	N/A	X
Total Non-Discounted Non-Investment Costs	totCostsNonDiscNonInv	List of floats	N/A	X
Total Discounted Non-Investment Costs	totCostsDiscNonInv	List of floats	N/A	X
Total Non-Discounted Non-Investment Benefits	totBenefitsNonDiscNonInv	List of floats	N/A	X
Total Discounted Non-Investment Benefits	totBenefitsDiscNonInv	List of floats	N/A	X
Total Direct Costs	totCostDir	List of floats	N/A	X
Total Discounted Direct Costs	totCostsDirDisc	List of floats	N/A	X
Total Direct Benefits	totBenefitsDir	List of floats	N/A	X
Total Discounted Direct Benefits	totBenefitsDirDisc	List of floats	N/A	X
Total Indirect Costs	totCostsInd	List of floats	N/A	X
Total Discounted Indirect Costs	totCostsIndDisc	List of floats	N/A	X
Total Indirect Benefits	totBenefitsInd	List of floats	N/A	X
Total Discounted Indirect Benefits	totBenefitsIndDisc	List of floats	N/A	X
Total Externality Costs	totCostsExt	List of floats	N/A	X
Total Discounted Externality Costs	totCostsExtDisc	List of floats	N/A	X
Total Externality Benefits	totBenefitsExt	List of floats	N/A	X
Total Discounted Externality Benefits	totBenefitsExtDisc	List of floats	N/A	X

Methods:

add_base_cost

Purpose: Adds a cost flow element by element to the total costs flow

Input: self, flow

Output: cost flow

add_base_benefits

Purpose: Adds a benefit flow element by element to the total benefits flow

Input: self, flow

Output: benefit flow

add_invest_cost

Purpose: Adds an investment cost flow element by element to the total investment costs flow

Input: self, flow

Output: investment cost flow

add_invest_benefits

Purpose: Adds an investment benefit flow element by element to the total investment benefits flow

Input: self, flow

Output: investment benefits flow

add_non_invest_cost

Purpose: Adds a non-investment cost flow element by element to the total non-investment costs flow

Input: self, flow

Output: non-investment cost flow

add_non_invest_benefits

Purpose: Adds an investment benefit flow element by element to the total non-investment benefits flow

Input: self, flow

Output: non-investment benefits flow

add_direct_cost

Purpose: Adds a direct cost flow element by element to the total direct costs flow

Input: self, flow

Output: direct cost flow

add_direct_benefit

Purpose: Adds a direct benefit flow element by element to the total direct benefits flow

Input: self, flow

Output: direct benefit flow

add_indirect_cost

Purpose: Adds an indirect cost flow element by element to the total indirect costs flow

Input: self, flow

Output: indirect cost flow

add_direct_benefit

Purpose: Adds an indirect benefit flow element by element to the total indirect benefits flow

Input: self, flow

Output: indirect benefit flow

add_external_cost

Purpose: Adds an external cost flow element by element to the total external costs flow

Input: self, flow

Output: external cost flow

add_external_benefit

Purpose: Adds an external benefit flow element by element to the total external benefits flow

Input: self, flow

Output: external benefit flow

add

Purpose: Performs summations to generate all total flows

Input: self, Bcn

Output: object updated with total flows values

__repr__

Purpose: Returns the Cash Flow's dictionary as a string

Input: self

Output: dict

OptionalCashFlow

Purpose: Calculates and stores total cash flows for tag and non-monetary flows

`__init__` input: self, altID, tag, units, studyPeriod

Attributes: Table 3-9

Table 3-9 Optional Cash Flow Variable Details

Variable	Var Name	Var Type	Format/Values	Req
Alternative ID	altID	Int	N/A	X
Tag	tag	String	N/A	X
Total Monetary Flow for Tag	totTagFlowDisc	List of Floats	N/A	X
Total Non-monetary flow for Tag	totTagQ	List of Floats	N/A	X
Units for the Tag	units	String	N/A	X

Methods:

add

Purpose: performs additions required to generate flows for the object

Input: self, bcn, flow

Output: object updated with total flows values

__repr__

Purpose: Returns the Optional Cash Flow's dictionary as a string

Input: self

Output: dict

3.2.2.3 SensitivitySummary (SensitivitySummarySerializer)

Purpose: Stores sensitivity output generated by the API

Required libraries: None

3.2.2.3.1 Functions

None

3.2.2.3.2 Classes

SensitivitySummary

Purpose: Stores sensitivity output generated by the API

`__init__` input: self, globalVarBool, bcnObj, varName, diffType, diffVal, diffSign,
measure_summaries

Attributes: Table 3-10

Table 3-10 SensitivitySummary Variable Details

Variable	Var Name	Var Type	Format/Values
Global Variable Boolean	globalVarBool	Boolean	True/False
Name of altered BCN	bcnObj	String	N/A
BCN variable being altered	varName	String	Valid entries: Same as Sensitivity.varName
Nature of the alteration to the BCN value	diffType	String	Valid entries are: “Gross”, “Percent”
Amount the BCN value is changed	diffValue	Float	Can be negative or positive
Sign of change	diffSign	Int	-1/1
Total Benefits Flow	totalBenefits	List of Floats	N/A
Total Costs Flow	totalCosts	List of Floats	N/A
Total Investment Costs	totalCostsInv	List of Floats	N/A
Total Non-Investment Costs	totalCostsNonInv	List of Floats	N/A
Total Subtype Flows	totalSubtypeFlows	List of List of Floats	N/A
Net Benefits	netBenefits	List of Floats	N/A
Net Savings	netSavings	List of Floats	N/A
Savings to Investment Ratio	SIR	List of Floats	N/A
Internal Rate of Return	IRR	List of Floats	N/A
Adjusted Internal Rate of Return	AIRR	List of Floats	N/A
Simple Payback Period	SPP	List of Floats	N/A
Discounted Payback Period	DPP	List of Floats	N/A
Benefit cost Ratio	BCR	List of Floats	N/A
Quantity Sum	quantSum	List of List of Floats	N/A
Quantity Units	quantUnits	List of List of Floats	N/A
MARR	MARR	Float	N/A
Change in Quantity	deltaQuant	List of List of Floats	N/A
Net savings per change in quantity	nsPercQuant	List of List of Floats	N/A
Net savings per change in percent quantity	nsDeltaQuant	List of List of Floats	N/A
Net savings elasticity	nsElasticityQuant	List of List of Floats	N/A

__str__

Purpose: Allows object variables to be printed using a print statement. Only used for debugging purposes

Input: self

Output: list of values for an instance of the object class

3.2.2.4 Output (OutputSerializer)

Purpose: Stores all outputs generated by the API in a single object

Required libraries: None

3.2.2.4.1 Functions

None

3.2.2.4.2 Classes

Output

Purpose: Stores all Outputs generate by the API in a single object

__init__ input: self, **kwargs

Attributes: Dependent on the objToReport variable defined by the user. Must include the CashFlows object and optionally the AlternativeSummary and/or the SensitivitySummary objects.

Methods: None

3.2.3 Modules

All modules are organized into directories including the following python files:

__init__ which makes the module importable.

admin which is used to register models and is not used for any of the modules in E3.

apps which contains the executable code for the module. All subsequent sections refer to code in the modules respective *app* file.

models which allows for the creation of models and is not used for any of the modules in E3.

tests which stores unit tests for the module. Tests for modules were run separately from these files so they are blank.

views which is used to define graphical GUI elements for the module and is not used for any of the modules in E3.

3.2.3.1 cashflows

Purpose: Organize calculations required to obtain the cash flows for Required and Optional objects

Required libraries: math, decimal, typing, API.objects, API.registry, API.variables

3.2.3.1.1 Functions

present_value

Purposes: Calculates the present value of a future cash flow

Inputs: v (value), d (discount rate), t (time), timestep_comp

Outputs: present value of v

discount_values

Purpose: discounts an entire list of values

Input: rate, value_list, timestep_comp

Output: list of discounted values

calculate_values

Purpose: Calculates the non-discounted values for the BCN over the study period

Input: bcn, study_period, quantities

Output: list of non-discounted values

get_recur_generator

Purpose: Selects the appropriate method of recurrence calculation based on value in the recurValRate attribute

Input: bcn

Output: var_value

year_by_year

Purpose: Calculates variability rates year by year

Input: intial, recur_end_date, var_value_list

Output: var_value_list

single_value

Purpose: calculates the var_value for a non-compounding BCN

Input: self

Output: list of recurVarValues

var_value

Purpose: Calculates the var_value for a compounding BCN

Input: bcn

Output: list of compounded var_values

quantitie_generator

Purpose: Calculates quantities over the study period

Input: self, bcn

Output: list of escalation values for quantity

calculate_quantities

Purpose: Calculates quantities, including escalation rates if present, by timestep

Input: bcn, study_period

generator_base

Purpose: General purpose function to form a generator that repeats cash flow calculations

Input: bcn, study_period

Output: input calculation repeated over the study period

remaining_life

Purpose: calculates the remaining life of a BCN with a service life for the residual_value method

Input: self, study_period

Output: remaining life

residual_value

Purpose: Calculates the residual value for a BCN

Input: bcn, study_period, values

Output: residual value

cash_flows

Purpose: Discounts the BCN and returns all intermediate and final values

Input: bcn, study_period, rate, timestep_comp

Output: quantities, values, discounted_values

3.2.3.1.2 Classes

CashFlowConfig

Purpose: Configures cashflow module so it is callable, importable, and defines its return values

Attributes:

name: compute.cashflow
 verbose_name: E3 Cash Flow Generator
 output: internal:cash-flows

Methods:

run

Purpose: Initiates calculations related to the module when called by the ModuleGraph in registry

Input: self, base_input (optional)

Output: Dictionary of all BCN cash flows by timestep

3.2.3.2 measures

Purpose: Organize calculations required to obtain the cash flows for Required and Optional objects

Required libraries: typing, rest_framework.fields, API.objects, API.registry, compute.objects, compute.serializers

3.2.3.2.1 Functions

calculate_alternative_summaries

Purpose: Calculates quantities over the study period

Input: analysis, required_flows, optional_flows, alternatives

Output: List of AlternativeSummary objects

3.2.3.2.2 Classes

AlternativeSummaryConfig

Purpose: Configures measures module so it is callable, importable, and defines its return values

Attributes:

name: compute.cashflow

verbose_name: E3 Cash Flow Generator
 depends_on: ["FlowSummary", "Optional Summary"]
 output: MeasureSummary
 serializer: ListField(child=AlternativeSummarySerializer, required=False)

Methods:

run

Purpose: Initiates calculations related to the module when called by the ModuleGraph in registry

Input: self, base_input (optional)

Output: List of all AlternativeSummary objects

3.2.3.3 optional

Purpose: Calls calculations required to obtain the OptionalCashFlow objects

Required libraries: rest_framework.fields, API.registry, compute.objects, compute.serializers

3.2.3.3.1 Functions

create_empty_tag_flows

Purpose: Generates an empty cash flow to store cash flows

Input: user_input

Output: results

calculate_tag_flows

Purpose: Calculates cash flows by timestep for each tag and stores them in a list

Input: flow, user_input, bcnObjects (optional)

Output: list of all tagged flows

3.2.3.3.2 Classes

OptionalCashFlowConfig

Purpose: Configures optional module so it is callable, importable, and defines its return values

Attributes:

name: compute.optional

verbose_name: E3 Optional Cash Flow Generator
 depends_on: ["internal:cash-flows"]
 output: OptionalSummary
 serializer: ListField(child=OptionalCashFlowSerializer, required=False)

Methods:

run

Purpose: Initiates calculations related to the module when called by the ModuleGraph in registry

Input: self, base_input (optional)

Output: List of all AlternativeSummary objects

3.2.3.4 required

Purpose: Calls calculations required to obtain the CashFlows objects

Required libraries: rest_framework.fields, API.registry, compute.objects, compute.serializers

3.2.3.4.1 Functions

calculate_required_flows

Purpose: Calls calculations required to generate RequiredCashFlows objects

Input: bcnObjects, studyPeriod, cashFlows

Output: list of RequiredCashFlows objects

3.2.3.4.2 Classes

RequiredCashFlowConfig

Purpose: Configures required module so it is callable, importable, and defines its return values

Attributes:

name: compute.required
 verbose_name: E3 Required Cash Flow Generator
 depends_on: ["internal:cash-flows"]
 output: FlowSummary
 serializer: ListField(child=RequiredCashFlowSerializer, required=False)

Methods:

run

Purpose: Initiates calculations related to the module when called by the ModuleGraph in registry

Input: self, base_input, dependencies (optional)

Output: List of all RequiredCashFlowobjects

3.2.3.5 sensitivity

Purpose: Calls calculations required to obtain the OptionalCashFlow objects

Required libraries: rest_framework.fields, API.registry, compute.cashflow.apps, compute.objects, compute.optional.apps, compute.required.apps, compute.measures.apps, compute.serializers, numpy

3.2.3.5.1 Functions

None

3.2.3.5.2 Classes

SensitivityConfig

Purpose: Configures required module so it is callable, importable, and defines its return values

Attributes:

name: compute.sensitivity

verbose_name: E3 Sensitivity Generator

depends_on: ["internal:cash-flows"]

output: SensitivitySummary

serializer: ListField(child=SensitivitySummarySerializer, required=False)

Methods:

run

Purpose: Initiates calculations related to the module when called by the ModuleGraph in registry

Input: self, base_input, dependencies (optional)

Output: List of SensitivitySummaryObjects

3.2.4 Support Libraries – Calculations

Apart from the support libraries listed in the remainder of Section 3.2.4, there are multiple `__init__` files located throughout the file structure. These exist to make libraries within the folders that contain an `__init__` file importable by other files. As such they are not discussed beyond this description, nor are they broken out by individual file.

3.2.4.1 tasks

Purpose: initiates and orchestrates all calculations upon receiving user generated input

Required libraries: celery, API.objects, API.registry, API.serializers.OutputSerializer, copy

3.2.4.1.1 Functions

analyze

Purpose: Equivalent to main, initializes and runs all calculations

Input: user_input

Output: API output

3.2.4.1.2 Classes

None

3.2.4.2 registry

Purpose: initiates and orchestrates all calculations upon receiving user generated input

Required libraries: typing, django.apps, rest_framework.fields, API.objects, API.serializers.OutputSerializer

3.2.4.2.1 Functions

None

3.2.4.2.2 Classes

E3ModuleConfig

Purpose: Configures module format and allows modules to be run

Attributes:

app_name: name of the app running the module

app_module: name of the module being run

Methods:

run

Purpose: Generic run call for a module

Input: self, base_input, dependencies (optional)

Output: None

Node

Purpose: Represents a module node in the dependency graph

Attributes:

 dependency: list of dependent nodes

 module: name of the module being run

Methods:

add_dependency

Purpose: Adds a node as a dependency

Input: self, node

Output: new Node

run

Purpose: Run call for a module associated with the specific node, skips calculation and returns a value if one is already stored in the cache

Input: self, base_input, dependencies (optional)

Output: Result

Singleton

Purpose: Metaclass to turn a class into a singleton

Attributes:

 None

Methods:

__call__

Purpose: Converts class to a singleton

Input: self, *args, **kwargs

Output: singleton

ModuleGraph

Purpose: Singleton class to maintain the entire module dependency class

Attributes:

Nodes, open_nodes

Methods:

add_module

Purpose: Adds module to the dependency graph

Input: self, module

Output: updated dependency graph

run

Purpose: Gets the requested output object from its associated module

Input: self, base_input, cache

Output: Output object

__repr__

Purpose: Returns the Nodes of the Module Graph as a string

Input: self

Output: string representation of Nodes

3.2.4.3 Variables

Purpose: Defines max digits and decimal places, as well as the API defined types *CostType*, and *FlowType*. Defines VAR_RATE_OPTIONS and NUM_ERRORS_LIMIT.

Required libraries: decimal

3.2.4.3.1 Functions

None. This file is a series of commands without a *def* block. See code for specifics

3.2.4.3.2 Classes

None

3.2.5 Support Libraries – API

API libraries are treated differently in this document. Since many of them are lists of commands they have no functions in the sense that they are defined by a *def* block. For that reason, the API support libraries are only presented by their purpose in the code. This section is organized by the directory (defined in the subsection headings) that the files are in using E3/es_django as the base directory. Directories labeled tests are not included as they only contain the tests for validating the code and play no role in calculation or API functionality. Any *migrations* directories are also omitted as they are autogenerated by Django.

3.2.5.1 /API

admin: Allows definition of models. Not used in current version of E3.

celery: Sets up app and the app environment and app configuration

urls: Defines routes for the built in API GUI

views: A collection of classes used to present output to the user based on what page they are on and the status of a request

3.2.5.2 /e3_django

asgi: ASGI configuration for e3_django project

settings: Django settings for e3_django project

urls: e3_django URL configuration

wsgi: wsgi configuration for e3_django project

3.2.5.3 /E3_local

List of files automatically generated to allow E3 to run locally in a virtual environment. Full details omitted as this is autogenerated by the requirements file and the creation of a virtual environment.

3.2.5.4 /frontend

Collection of files and directories used to generate the built in GUI for E3 and connect it to the code to run the app. */static* and */templates* contain the code that generates the GUI, while the files in */frontend* allow the app to authenticate, add users, and receive requests.

3.2.6 Additional Files

Some other files worth noting are presented here.

E3/e3_django/requirements.txt: Lists all required libraries and the specific version that must be installed for the API to function. Is callable to install the entire list when creating a virtual environment.

E3/e3_django/LICENSE: NIST software license:

This data/work was created by employees of the National Institute of Standards and Technology (NIST), an agency of the Federal Government. Pursuant to title 17 United States Code Section 105, works of NIST employees are not subject to copyright protection in the United States. This data/work may be subject to foreign copyright.

The data/work is provided by NIST as a public service and is expressly provided “AS IS.” NIST MAKES NO WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED

WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT AND DATA ACCURACY. NIST does not warrant or make any representations regarding the use of the data or the results thereof, including but not limited to the correctness, accuracy, reliability or usefulness of the data. NIST SHALL NOT BE LIABLE AND YOU HEREBY RELEASE NIST FROM LIABILITY FOR ANY INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, AND THE LIKE), WHETHER ARISING IN TORT, CONTRACT, OR OTHERWISE, ARISING FROM OR RELATING TO THE DATA (OR THE USE OF OR INABILITY TO USE THIS DATA), EVEN IF NIST HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

To the extent that NIST may hold copyright in countries other than the United States, you are hereby granted the non-exclusive irrevocable and unconditional right to print, publish, prepare derivative works and distribute the NIST data, in any medium, or authorize others to do so on your behalf, on a royalty-free basis throughout the world.

You may improve, modify, and create derivative works of the data or any portion of the data, and you may copy and distribute such modifications or works. Modified works should carry a notice stating that you changed the data and should note the date and nature of any such change. Please explicitly acknowledge the National Institute of Standards and Technology as the source of the data: Data citation recommendations are provided at <https://www.nist.gov/open/license>.

Permission to use this data is contingent upon your acceptance of the terms of this agreement and upon your providing appropriate acknowledgments of NIST's creation of the data/work.

Note that if any changes are made by a third party without NIST's consent this LICENSE no longer applies. Any such software may still be used and distributed, but the LICENSE itself will be void and should be deleted from directory of the altered code.

E3/e3_django/README.md: A useful document that provides guidance on how to run E3 locally.

3.3 Planned Changes in Development Tools and File Structure

E3 is currently being redeveloped in the Java programming language. The decision to was made to facilitate optimal processing of requests for uncertainty analysis. The Java version represents a major change in overall structure and is intended to be the final code base for E3. The E3-python code (v1) provides valid calculations and is being preserved for those who may prefer to use it, however there are no plans to add uncertainty calculations to the E3-python (v1) code base nor provide support for it beyond patching security issues that may arise.

3.3.1 Java

The re-design of E3 will have several components. First, the main calculation engine will be implemented in a modular fashion using Function Reactive Programing (FRP) to facilitate expandability and optimization. FRP allows for a specific calculation model to be dynamically constructed that can optimally update itself for altered inputs in analysis types like sensitivity and uncertainty. Different analysis types will be implemented as separate modules, which allows for the engine to be easily expanded later for specific tools.

Second, the Web server will be switched to the Spring Web Framework. Spring is a popular Java web framework that is mature and has many development options. This allows for developers to easily extend the Web-side of E3 if they need to accommodate different protocols or use-cases.

Finally, the calculation engine itself will be entirely separate from the E3 web server so it can be integrated directly into other applications. Other languages that use the JVM such as Kotlin and Scala will be able to use E3 naively, but other languages outside of the JVM may require additional work. A command-line interface (CLI) implementation can be easily made to allow other programs to invoke E3 through the operating system.

3.3.2 JavaScript Submission Library

Additionally, a JavaScript library is being developed that will simplify the building and sending of analysis requests to an E3 instance. This library is not strictly necessary to use E3 since REST requests can still be manually constructed and sent. However, it can break up requests into easily understandable and re-usable parts that can aid in application development. A builder pattern is used to store information until an analysis is requested, which then triggers the library to take the stored information, infer any information it can (e.g., IDs), send constructed JSON request to E3, and return the result.

This library is intended to be used to create web applications, but similar libraries could be made for use in desktop applications as well.

4 Using E3

While E3 can take input directly if the underlying code is run on a computer locally, it is intended to be called by a user-created Web interface. At present the API requires registration, at which time a security key will be provided. The API checks for a key before accepting any requests from a client and, upon accepting, proceeds to pass data to the portions of the code responsible for calculation. This data is then sent back to the client to be formatted by the client-side interface.

4.1 Overview

As E3 is currently a Beta version, specifics on the procedure for calling on the API from a Web interface, as well as guidance on how to handle certain situations, is in flux. This section is forthcoming upon finalization of base functionality.

4.1.1 Information Exchange Formats

E3 takes input and produces output in the form of JavaScript Oriented Notation (JSON) strings. This provides a consistent format for input and provides a consistent output type to allow developers to know exactly how communication with the API will proceed between the API and the interface calling it. Internally data is converted to objects and processed through serializers to allow calculation and rapid transition to and from the JSON format.

4.2 Sending a Request to E3 API Web Server

The process for sending a request to the E3 API server is as follows:

First, the JSON object needs to be constructed by the client. The format for a valid JSON request is specified below. Note that some result objects take much longer to calculate than others, which means responses may not be instantaneous.

Second, send the JSON object via a POST request to the “/api/v1/analysis/” endpoint of the E3 server with the API key in the header in the form:

“Authorization: Api-Key YOUR_KEY_HERE”

If the JSON object sent is invalid, an error will be sent back containing information about what is incorrect. Errors will also be sent back if something went wrong with the calculations. There may be errors that cannot be detected pre-processing, such as particular input parameters that cause divide by zero errors. Error messages will provide as much information as possible to aid in debugging requests. Once the server finishes the calculations for a request, the response is sent back as another JSON object, which the client can then interpret and display as they wish.

4.3 Running E3 API Locally

Instead of using the E3 web API, a user may prefer to download the code base to use locally. In such a case, please refer to the README.md file in the E3 GitHub repository.

There are several potential reasons for preferring this option, including the following:

- Expand computing power of E3
- Develop a desktop application using E3
- Add or edit capabilities and features in E3

The process for running E3 locally is as follows:

Run with Docker

Ensure that docker and docker-compose are installed on your system.

Run with docker-compose:

```
$ docker-compose build
```

```
$ docker-compose up
```

This will run the django development server at <http://localhost:8000/>. PostgreSQL will be used as the database and RabbitMQ and a worker will be created for API requests.

Note: If you are running it for the first time, you need to create a superuser:

Run the following command:

```
docker ps
```

From the output list, find your CONTAINER ID for the IMAGE e3_api. Then run the following commands:

```
docker exec -it [CONTAINER ID] bash
```

```
python manage.py makemigrations
```

```
python manage.py migrate
```

```
python manage.py createsuperuser
```

Enter your email and password when prompted.

Go to localhost:8000/login, and login with your email and password.

Obtain your unique API key (Please COPY and store somewhere safe - you will only be shown this once).

Now you can send requests to:

`http://localhost:8000/api/v1/analysis/`

with an authorization HTTP header in the form:

“Authorization: Api-Key YOUR_KEY_HERE”

Getting Started

Clone the repository from Github.

```
$ git clone https://github.com/usnistgov/E3.git
```

```
$ cd e3_django
```

Run Locally

Activate the virtualenv for your project.

Install project dependencies:

```
$ pip install -r requirements.txt
```

Then simply apply the migrations:

```
$ python manage.py migrate
```

You can now run the development server:

```
$ python manage.py runserver
```

4.4 Guidance

Along with providing the details on the E3 code, it is beneficial to provide front-end developers guidance on how to map what their users will provide in their application to the appropriate E3 file format as well as how to interpret the E3 output file and how the results map to their desired performance measures. Each application will have their own unique set of information that could be using different terminology, desired results outputs, and potential irregular calculation requests. This section provides guidance by analysis type for both input and output files as well as identified edge cases that require careful consideration in designing their application as well as sending requests to and receiving results from E3.

4.4.1 General Guidance

There are several items of guidance that are independent of the type of analysis to be completed. Additional items will be added periodically as they are identified.

First, if a variable is not used in the analysis or required to calculate another variable required in the analysis, then it can be “null” or excluded completely, including the following:

- Analysis Object
 - interestRate (optional)
 - incomeRateFed (optional)
 - incomeRateOther (optional)
 - location (optional)
- BCN Object
 - bcnLife (if rvBool = false)
 - recurInterval (if recurBool = false)
 - recurVarRate (if recurBool = false)
 - recurVarValue (if recurBool = false)
 - recurEndDate (if recurBool = false or if recurrence through end of study period)
 - quantVarRate (if quant does not vary)
 - quantVarValue (if quant does not vary)
 - quantUnit (optional)

Second, residual values for an investment cost can be calculated using two approaches. (1) A user can specify for E3 to calculate the residual value of an investment cost by setting rvBool = true, which tells E3 to calculate the residual value of the investment using a linear depreciation approach based several BCN parameters (e.g., investment costs, initial and future occurrences, life of the investment, study period of the analysis) to calculate and include the residual value in the analysis automatically. This approach simplifies the inputs required by the user, but it limits transparency in the results and flexibility of the residual value calculation. (2) A user can define a separate BCN object that explicitly defines the residual value. In this case, the user must input all the necessary information, including the value and time of occurrence. The user must set rvBool = true to tell E3 to calculate the residual value and rvOnly = true to tell E3 that the BCN is a residual value object. The valuePerQ should be provided as a positive value. This approach is more transparent and allows the user to use whatever method to calculate the residual value, and not just linear depreciation approach used in (1).

Third, E3 currently does not allow for an E3 request to include a combination of real and nominal denominated BCN object values. Instead, E3 assumes that all BCN objects are

provided in the same term as defined in the Analysis Object (outputRealBool parameter). The bcnRealBool value is not currently checked for consistency, but will be used in future versions of E3 to allow for providing a combination of real and nominal denominated BCN objects.

4.4.2 Guidance by Analysis Type

Due to the variety in the types of analysis required as well as varying terminology by topic area for which E3 can be leveraged for back-end economic calculations, it is beneficial to provide general guidance on how to map benefits, costs, and non-monetary values to E3 input file formatting. Additional guidance on other types of analysis will be added as validation of those analysis types are completed. Note that this guidance is only focused on the alternative summary results and does not discuss annual cash flows or non-monetary quantities.

4.4.2.1 Guidance for LCCA

LCCA is commonly used to evaluate the cost-effectiveness of an investment. Two examples for the same project with different financing options are provided in this section, one completely funded completely upfront and another financed with a small down payment and fixed monthly loan payments.

4.4.2.1.1 Inputs – All Real Values

The example below is based on Example 5-1 from Section 5.2 in Handbook 135 [15]. Assume a federal agency is deciding whether to install a cheaper, conventional heating, ventilation, and air conditioning (HVAC) system or a more expensive, high-efficiency HVAC system at its facility. The study period is assumed to be the lifetime of the HVAC system at 20 years starting from the start of operating the new system, which is assumed to be January 1, 2022. The assumed real discount rate, MARR, and reinvestment rate are 3.0 % and the general rate of inflation is expected to be 2.3 %. The data is provided on a yearly basis, which will be used as the timestep for the calculations. The results we are interested in are only the summary measure results (Total LCC, NS, SIR, AIRR, and payback period). Based on this information, the Analysis Object would use the following values in Table 4-11.

Table 4-11 Analysis Values

Var Name	Format/Values	Req
analysisType	LCCA	X
projType	Buildings	
objToReport	[MeasureSummary]	X
studyPeriod	20	X
baseDate	2022-01-01	X
serviceDate	2022-01-01	
timestepVal	Year	X
timestepComp	EndOfYear	X
outputRealBool	True	X
interestRate		
dRateReal	0.030	X
dRateNom		
inflationRate	0.023	
Marr	0.030	X
reinvestRate	0.030	X
incomeRateFed		
incomeRateOther		
noAlt	2	X
baseAlt	0	X
location		

Table 4-12 provides the current and future costs for both alternatives. All costs are provided in constant dollars.

Table 4-12 Example Costs

Cost Category	Cost Item	Occurrence(s)	Costs by Alternative	
			Conventional	High Efficiency
Initial Investment	HVAC System	Year 0	\$103 000	\$130 000
Capital Replacement	Fan	Year 15	\$12 000	\$12 500
Operational	Electricity	Annual	\$30 000	\$24 000
	Maintenance	Annual	\$7000	\$8000
Residual Value	Capital Equipment	Year 20	-\$3500	-\$3700

The maintenance cost is assumed to be constant in real terms while the electricity price is \$0.12/kWh and remains constant over the study period. Electricity consumption is assumed to be constant year-over-year (300 000 kWh for the conventional and 240 000 kWh for the high efficiency system). The costs for the conventional system map to E3 BCN object values as follows in Table 4-13:

Table 4-13 BCN Object Values - Conventional

VarName	HVAC System	Fan Replacement	Electricity	Maintenance	Residual Value
altID	0	0	0	0	0
bcnID	0	1	2	3	4
bcnType	Cost	Cost	Cost	Cost	Cost
bcnSubType	Direct	Direct	Direct	Direct	Direct
bcnName	HVAC System	Fan Replacement	Electricity	Maintenance	Residual Value
bcnTag	Initial Investment	Capital Replacement	Energy	OMR	Capital Salvage Value
initialOcc	0	12	1	1	20
bcnRealBool	true	true	true	true	true
bcnInvestBool	true	true	false	false	True
bcnLife	20	12	Null	Null	Null
rvBool	false	false	false	false	True
rvOnly	false	false	false	false	True
recurBool	false	false	true	true	False
recurInterval	Null	Null	1	1	Null
recurVarRate	Null	Null	Percent Delta Timestep X-1	Percent Delta Timestep X-1	Null
recurVarValue	Null	Null	0	0	Null
recurEndDate	Null	Null	20	20	Null
valuePerQ	103000	12000	0.12	7000	3500
quant	1	1	250000	1	1
quantVarRate	Null	Null	Null	Null	Null
quantVarValue	Null	Null	Null	Null	Null
quantUnit	1	1	kWh	1	1

The costs for the high-efficiency system map to E3 BCN object values as follows in Table 4-14:

Table 4-14 BCN Object Values – High-Efficiency

VarName	HVAC System	Fan Replacement	Electricity	Maintenance	Residual Value
altID	0	0	0	0	0
bcnID	0	1	2	3	4
bcnType	Cost	Cost	Cost	Cost	Cost
bcnSubType	Direct	Direct	Direct	Direct	Direct
bcnName	HVAC System	Fan Replacement	Electricity	Maintenance	Residual Value
bcnTag	Initial Investment	Capital Replacement	Energy	OMR	Capital Salvage Value
initialOcc	0	12	1	1	20
bcnRealBool	1	1	1	1	1
bcnInvestBool	1	1	0	0	1
bcnLife	20	12	Null	Null	Null
rvBool	false	false	false	false	true
rvOnly	false	false	false	false	true
recurBool	0	0	1	1	0
recurInterval	Null	Null	1	1	Null
recurVarRate	Null	Null	Percent Delta Timestep X-1	Percent Delta Timestep X-1	Null
recurVarValue	Null	Null	0	0	Null
recurEndDate	Null	Null	20	20	Null
valuePerQ	130000	12500	0.12	8000	-3700
quant	1	1	240000	1	
quantVarRate	Null	Null	Null	Null	Null
quantVarValue	Null	Null	Null	Null	Null
quantUnit					

Note that if the user is not concerned about the quantity of electricity consumed, the user could input total annual costs as the ValuePerQ and change the Quantity to one (1) and still obtain the same results.

The results in Table 4-15 show that the high efficiency system leads to net savings of \$47 147, an SIR of 2.73, and AIRR of 8.3 %.

Table 4-15 LCCA Example Results

Net Present Value	Conventional	High Efficiency
Initial Investment Cost	\$103 000	\$130 000
Capital Replacement	\$8417	\$8767
Residual Value	-\$1938	-\$2049
Electricity	\$446 324	\$357 059
OMR	\$104 142	\$119 019
Total Life Cycle Cost	\$659 945	\$612 798
Net Savings		\$47 147
SIR		2.73
AIRR		8.3 %

The results reports generated by E3 are independent of the type of analysis. LCCA will be focused on a subset of the results. The Measure Summary reports all the variables in Table 4-16. Those **bolded** are measures used in LCCA.

Table 4-16 Alternative Summary Object Results Values

Variable	VarName	Format
Alternative ID	altID	Digit
Total Benefits	totalBenefits	Float
Total Costs	totalCosts	Float
Total Costs – Investment	totalCostsInv	Float
Total Costs – Non-investment	totalCostsNonInv	Float
Net Benefits	netBenefits	Float
Net Savings	netSavings	Float
Savings-to-Investment Ratio	SIR	Float
Internal Rate of Return*	IRR	Float
Adjusted Internal Rate of Return	AIRR	Float
Simple Payback Period	SPP	Float
Discounted Payback Period	DPP	Float
Benefit-Cost Ratio	BCR	Float
Quantity by Tag	quantSum	List of Floats
Quantity Units by Tag	quantUnits	List of Strings
MARR	MARR	Float
Delta Quantity by Tag	deltaQuant	List of Floats
Delta Net Savings by Tag	nsDeltaQuant	List of Floats
Net Savings Percent Quantity by Tag	nsPercQuant	List of Floats
Net Savings Elasticity by Quantity by Tag	nsElasticityQuant	List of Floats
* Included if IRRSummary is included in ObjectsToReport		

4.4.2.1.2 Inputs – Combination of Real and Nominal Values

Capital investments are commonly financed, at least in part, due to funding constraints. The same example from Section 4.4.2.1.1 can be modified to replace the initial investment costs with a financing option (down payment and monthly loan payments). Assume that the conventional system project is funded using a loan with the following terms:

- 15-year loan
- 10 % down payment (\$10 300)
- \$700 monthly payments

The BCN objects are identical for all costs except that the initial investment for “HVAC System” is decreased from \$103 000 to \$10 300. An additional BCN is necessary to cover the loan payments (12 payments of \$700 annually). Currently, E3 does not allow for an E3 request to include a combination of real and nominal denominated BCN object values. Instead, E3 assumes that all BCN objects are provided in the same term as defined in the Analysis Object (outputRealBool parameter). Therefore, an additional step

is required to convert the nominal/current loan payments to real/constant dollars. This is accomplished by including an escalation rate equal to the negative of the inflation rate to convert the nominal values to real values as shown in Table 4-17.

Table 4-17 BCN Loan Object Values

VarName	Loan Payments
altID	0
bcnID	10
bcnType	Cost
bcnSubType	Direct
bcnName	Loan Payments
bcnTag	Financing
initialOcc	1
bcnRealBool	true
bcnInvestBool	False
bcnLife	Null
rvBool	False
rvOnly	False
recurBool	True
recurInterval	1
recurVarRate	Percent Delta Timestep X-1
recurVarValue	[-0.023]
recurEndDate	15
valuePerQ	700
quant	12
quantVarRate	Null
quantVarValue	Null
quantUnit	Null

4.4.2.1.3 Inputs – Nominal Values

A “current dollar” or nominal dollar analysis will set outputRealBool = false for the Analysis Object and bcnRealBool = false for all BCN objects. The analysis will use the nominal discount rate and general rate of inflation provided to calculate the results. An example using nominal values is provided in Section 4.5.1.1.2.

4.4.2.2 Guidance for BCA

BCA is a method that estimates all the benefits and costs associated with a decision. A key result from a BCA is the benefit-cost ratio (BCR), which takes the total benefits minus total non-investment costs and divides by the total investment costs (costs over which the user desires to maximize returns). The user determines which costs will be treated as investment costs, and therefore will influence the calculation of the BCR. For example, a user can choose whether to treat future costs as investment costs, which will impact both the numerator and denominator of the equation.

The example below is a simplistic analysis of the benefits and costs to a locality resulting from constructing a new monument. The study period is assumed to be the lifetime of the monument at 50 years is assumed to be January 1, 2022. The assumed real discount rate, MARR, and reinvestment rate are 3.0 % and the general rate of inflation is expected to be 2.3 %. The data is provided on a yearly basis, which will be used as the timestep for the calculations. The results we are interested in are only the summary measure results (Total Benefits, Total Costs, BCR, and IRR). Based on this information, the Analysis Object would use the following values in Table 4-18.

Table 4-18 Analysis Values

Var Name	Format/Values	Req
analysisType	BCA	X
projType		
objToReport	MeasureSummary	X
studyPeriod	50	X
baseDate	2022-01-01	X
serviceDate		
timestepVal	Year	X
timestepComp	EndOfYear	X
outputRealBool	true	X
interestRate		
dRateReal	0.030	X
dRateNom		
inflationRate	0.023	
Marr	0.030	X
reinvestRate	0.030	X
incomeRateFed		
incomeRateOther		
noAlt	2	X
baseAlt	0	X
location		

Table 4-19 provides the current and future benefits and costs for both alternatives. The baseline alternative is the status quo of not constructing the monument while the alternative to build the monument accounts for the incremental impacts relative to the status quo. Costs include the initial construction costs and annual OMR costs over the 50-year study period. The expected benefits are the free publicity from the media on the opening of the monument and an annually recurring increase in the economy from tourism resulting from the monument. All benefits and costs are provided in constant dollars.

Table 4-19 Example Benefits and Costs

Cost Category	Description	Alternative	
		Status Quo	Build Monument
Initial Investment Cost	Construct monument	\$0	\$100 000
OMR Cost	Maintain monument	\$0	\$1000
One-Time Benefit	Free publicity	\$0	\$50 000
Recurring Benefit	Increased tourism	\$0	\$5000

Table 4-20 maps the benefits and costs to E3 BCN objects.

Table 4-20 BCN Object Values - Conventional

VarName	Investment Cost – Status Quo	Construct Monument	Maintain Monument	Free Publicity	Increased Tourism
altID	0	1	1	1	1
bcnID	0	1	2	3	4
bcnType	Cost	Cost	Cost	Benefit	Benefit
bcnSubType	Direct	Direct	Direct	Direct	Externality
bcnName	Investment Cost – Status Quo	Construct Monument	Maintain Monument	Free Publicity	Increased Tourism
bcnTag	Investment Cost	Investment Cost	OMR Cost	Publicity	Tourism
initialOcc	0	0	1	1	1
bcnRealBool	true	true	true	true	true
bcnInvestBool	true	true	False	false	false
bcnLife	null	50	null	Null	Null
rvBool	false	false	false	false	false
rvOnly	false	false	false	false	false
recurBool	false	false	true	False	true
recurInterval	Null	Null	1	null	1
recurVarRate	Null	Null	Null	Null	Null
recurVarValue	Null	Null	Null	Null	Null
recurEndDate	Null	Null	Null	Null	Null
valuePerQ	0	100000	1000	50000	5000
quant	1	1	1	1	1
quantVarRate	Null	Null	Null	Null	Null
quantVarValue	Null	Null	Null	Null	Null
quantUnit	Null	1	1	1	1

The results in Table 4-21 show total net benefits of building the monument of \$51 463, a BCR pf 1.51, and an IRR of 7.3 %.

Table 4-21 BCA Example Results

Category	Net Present Value	Build Monument
Costs	Initial Investment Cost	\$100 000
	OMR	\$25 730
	Total	\$125 730
Benefits	Free Publicity	\$48 544
	Increased Tourism	\$128 649
	Total	\$177 193
Results	Net Benefits	\$51 463
	BCR	1.51
	IRR	7.3 %

The results reports generated by E3 are independent of the type of analysis. BCA will be focused on a subset of the results. The Alternative Summary reports all the variables in Table 4-22. Those **bolded** are measures used in BCA.

Table 4-22 Alternative Summary Object Results Values

Variable	VarName	Format
Alternative ID	altID	Digit
Total Benefits	totalBenefits	Float
Total Costs	totalCosts	Float
Total Costs – Investment	totalCostsInv	Float
Total Costs – Non-investment	totalCostsNonInv	Float
Net Benefits	netBenefits	Float
Net Savings	netSavings	Float
Savings-to-Investment Ratio	SIR	Float
Internal Rate of Return*	IRR	Float
Adjusted Internal Rate of Return	AIRR	Float
Simple Payback Period	SPP	Float
Discounted Payback Period	DPP	Float
Benefit-Cost Ratio	BCR	Float
Quantity by Tag	quantSum	List of Floats
Quantity Units by Tag	quantUnits	List of Strings
MARR	MARR	Float
Delta Quantity by Tag	deltaQuant	List of Floats
Delta Net Savings by Tag	nsDeltaQuant	List of Floats
Net Savings Percent Quantity by Tag	nsPercQuant	List of Floats
Net Savings Elasticity by Quantity by Tag	nsElasticityQuant	List of Floats
* Included if IRRSummary is included in ObjectsToReport		

4.4.2.3 Guidance for Profit Maximization

Profit maximization compares revenues to expenses to determine the net income the entity earns. This terminology is distinctly different than that used for both LCCA and BCA. Assume a simplified example with a potential capital investment that will increase productivity and, therefore, increase revenue. The decisionmaker can either maintain the

status quo or upgrade manufacturing equipment at an initial investment cost of \$1000 that will increase revenue by \$200 annually (in real terms) for the next 10 years (equipment's service life). Assume the real discount rate, MARR, and reinvestment rate are 3.0 % and the general rate of inflation is expected to be 2.3 %. The data is provided on a yearly basis, which will be used as the timestep for the calculations, and use End-of-Year discounting. The only results that the decisionmaker is interested in are change in revenue, change in costs, total profit, and IRR. Based on this information, the Analysis Object would use the following values in Table 4-23.

Table 4-23 Analysis Values

Var Name	Format/Values	Req
analysisType	Profit Maximization	X
projType	Manufacturing Process	
objToReport	MeasureSummary	X
studyPeriod	50	X
baseDate	2022-01-01	X
serviceDate		
timestepVal	Year	X
timestepComp	1	X
outputRealBool	True	X
interestRate		
dRateReal	0.030	X
dRateNom		
inflationRate	0.023	
Marr	0.030	X
reinvestRate	0.030	X
incomeRateFed		
incomeRateOther		
noAlt	2	X
baseAlt	0	X
location		

Table 4-24 provides the current and future costs for both alternatives. All costs are provided in constant dollars.

Table 4-24 Example Revenue and Costs

Category	Item	Occurrence(s)	Alternative	
			Status Quo	Upgrade
Initial Investment	Manufacturing Equipment	Year 0	0	1000
Additional Revenue	Additional Sales	Annually	0	200

E3 can be used two different ways to make the profit calculations. First, revenue could be treated as a negative cost, which leads to net savings to equal profit. Second, revenue could be treated as a benefit, which leads to net benefits to equal profit and total benefits to equal total additional revenue. Either approach will provide the same results, and the

user can decide which method works best for them. In this example, we will treat revenue as a negative cost and use life cycle cost analysis measures to report the results.

Table 4-25 shows the mapping of revenues and costs to E3 BCN objects.

Table 4-25 BCN Object Values – Profit Maximization using LCCA

VarName	Investment Cost	Investment Cost	Additional Revenue
altID	0	1	1
bcnID	0	1	2
bcnType	Cost	Cost	Benefit
bcnSubType	Direct	Direct	Direct
bcnName	Zero Cost	Initial Investment	Additional Revenue
bcnTag	Initial Investment	Equipment Upgrade	Additional Sales
initialOcc	0	0	1
bcnRealBool	true	true	true
bcnInvestBool	true	true	False
bcnLife	null	10	10
rvBool	false	False	false
recurBool	false	False	true
recurInterval	Null	Null	1
recurVarRate	Null	Null	Null
recurVarValue	Null	Null	Null
recurEndDate	Null	Null	Null
valuePerQ	0	1000	200
quant	1	1	1
quantVarRate	Null	Null	Null
quantVarValue	Null	Null	Null
quantUnit	1	1	1

The comparison to make in this example is non-investment costs (negative revenue) to investment costs (equipment upgrade) as shown in Table 4-26. The net savings, which is the negative of the change in costs relative to the status quo base case (\$0), provides the additional profit of upgrading the equipment over its 10-year service life.

Table 4-26 Profit Maximization Example Results

Category	Net Present Value	Upgrade
Profit	Revenue (Total Costs – Noninvestment)	-\$1706
	Total Costs - Investment	\$1000
	Net Savings = Profit	\$706
Returns	SIR	1.71
	AIRR	8.7 %
	IRR	15.1 %

4.4.2.4 Guidance for Monetary and Non-monetary Synergies and Trade-Offs

As the increase in sustainability has grown over the last decade, the scope of evaluations has broadened from solely monetary impacts to include non-monetary impacts that may

be difficult, or even impossible, to monetize. For example, the sustainable manufacturing standard includes metrics to determine the trade-offs between costs/profits and negative externalities (i.e., environmental impacts). These monetary/non-monetary comparisons could be applied to any analysis and, therefore, the example in this section will be simplified and generic to be applicable to different analysis types.

The example in Section 4.4.2.3 can be expanded to include a non-monetary value for environmental impacts (carbon emissions) associated with the equipment upgrade. Assume that the equipment has embodied carbon emissions associated with its manufacturing and installation of 300 kg of carbon dioxide equivalent (CO₂e) while it reduces operational energy consumption that leads to decreased emission of 100 kg CO₂e annually. Therefore, there are two additional BCN objects to include in the E3 input file, a one-time environmental impact in Year 0 and a recurring negative environment impact in Year 1 through Year 10 as shown in Table 4-27.

Table 4-27 BCN Object Values – Environmental Impacts

VarName	Embodied Emissions	Operational Energy Emissions
altID	1	1
bcnID	1	2
bcnType	Nonmonetary	Nonmonetary
bcnSubType	Externality	Externality
bcnName	Embodied Emissions	Operational Energy Emissions
bcnTag	Carbon Emissions	Carbon Emissions
initialOcc	0	1
bcnRealBool	True	True
bcnInvestBool	False	False
bcnLife	null	10
rvBool	False	False
recurBool	False	True
recurInterval	Null	1
recurVarRate	Null	Null
recurVarValue	Null	Null
recurEndDate	Null	Null
valuePerQ	1	1
quant	300	-100
quantVarRate	Null	Null
quantVarValue	Null	Null
quantUnit	Kg CO ₂ e	Kg CO ₂ e

The “deltaQuant” value for the “Carbon Emissions” tag is $300 - (100 * 10) = -700$ kg CO₂e. In this case there are synergies between profit (\$706) and environmental impacts (700 kg reduction), or a \$1.01 of additional profit per kg of carbon emissions reduction.

4.4.3 Other Output Results Options

There are other results that the user can report if they are determined to be useful, including an additional economic measure (IRR) and real and nominal cash flows. The IRR is reported within the Measures Summary but requires the inclusion of “IRRSummary” in the objects to report (objToReport) in the Analysis Object. The reason is that calculating the IRR requires a root finding algorithm that requires additional computing power and processing time.

Additionally, guidance thus far has focused on the Measures Summary results, which include summary metrics and aggregated costs and benefits by the standard BCN type-subtype categorization and optional BCN categorization by user defined BCN “tags.” E3 also provides detailed breakdown of nominal/non-discounted (“NonDisc”) and real/discounted (“Disc”) cash flows by both categorization options by including “FlowSummary” for the BCN categorization by type (cost, benefit, non-monetary) and subtype (direct, indirect, externality)⁴ and “OptionalSummary” for user defined BCN “tags” (e.g., initial investment, OMR, salvage value) in the objects to report (objToReport) in the Analysis Object.

4.4.4 Sensitivity and Uncertainty

Sensitivity and uncertainty analysis can be included for any type of analysis. Therefore, the examples provided in this section are generic to allow for them to apply to any use case of E3.

4.4.4.1 Sensitivity

Sensitivity analysis can be applied to both global and non-global parameters. Regardless of whether it’s a global or non-global variable, the last three values in the sensitivity object are straightforward to populate, where “varName” is the parameter to adjust, “diffType” is how the parameter value will be adjusted (by percentage or gross change and which direction(s) to make the change), and “diffVal” is the magnitude of the change to make to the parameter value.

A non-global variable, such as the value per quantity of a BCN object used by one or more alternatives, is defined within that specific BCN Object, and only impacts the calculations for the alternative(s) that use that BCN Object. Defining the sensitivity object for a non-global variable is straightforward because “globalVarBool” = false and there are values for “altID” and “bcnID” from BCN Object of interest.

⁴ The Flow Summary also provides a breakdown by investment and non-investment costs.

A global variable, such as the real discount rate, is defined in the Analysis Object and is used by all alternatives to make the economic calculations. Defining the sensitivity object for a global variable would set "globalVarBool" = true, "altID" = null, "bcnID" = null, and "bcnObj" = null because it does not apply to a single alternative/BCN. The "varName" is defined as the global parameter (e.g., "dRateReal"). Note that the only global parameter that can be changed is the discount rate used in the analysis (dRateReal or dRateNom). The next version of E3 will improve the input structure and global parameter options for sensitivity analysis.

4.4.4.2 Uncertainty

Uncertainty is not available in the current version of E3. The next release of E3, which is being redeveloped in Java, will include uncertainty capabilities. See Section 3.3 for additional details.

4.4.5 Edge Cases

Throughout the validation process numerous conditions that involve input values that require special handling (i.e., edge cases) have been identified in E3. This section documents those that have been identified. As more edge cases are found, they will be added to this section to assist E3 users.

4.4.5.1 E3 Inputs

There are input scenarios that can cause issues in E3 calculations if not designed correctly. Below is a list of those currently identified in E3:

- A user requests a BCA analysis with no Benefit BCN types.
- A BCN that is defined as recurring while only including one occurrence in the study period.
- A non-baseline alternative with no attached BCN objects might cause problems with calculating measures.
- The use of scientific notation for inputs must be formatted with the following notation: "1e3", and not "1X10^3", for a value of 1000.
- Some characters such, as white space and escape characters, may cause errors.

4.4.5.2 E3 Outputs

Currently only one edge case has been identified that the user should consider

- Root finding algorithms can have multiple solution values while E3 only provides a single value

4.5 E3 Web API Examples

The E3 web API is being tested with multiple use cases to validate the results for each of the features and capabilities. The initial use case was a web application that directly calls on E3 - Present Value of PhotoVoltaics – [PV]². Additional use cases include comparing results from JSON strings constructed from examples in existing software tools developed by AEO to the results generated directly from the software. Thus far this approach has been applied to two software tools: Building Life Cycle Cost (BLCC) [20] and Economic Decision Guide Software Online Tool (EDGe\$) [19].

BLCC is used to validate deterministic LCCA using both real and nominal discount rates, end-of-year and mid-year discounting conventions, alternatives with zero costs, and non-monetary unit comparisons.

EDGe\$ is used to validate deterministic analysis, sensitivity, and uncertainty for BCA using discrete end-of-year and continuous discounting conventions.

A sustainable manufacturing example from ASTM Standard E3200 is included that completes a LCCA of multiple machinery alternatives as well as evaluating “trade-off” measures for costs and environmental impacts.

The E3 JSON input files used for the BLCC, EDGe\$, and manufacturing examples are provided as a supplement resource to this publication in the E3 GitHub directory.

The [PV]² web application is used to validate the deterministic analysis, sensitivity, and uncertainty for LCCA, including all types of Direct Cost objects and results measures, for three alternatives. Additionally, Externality Costs and Non-Monetary BCNs are included for environmental impacts and the social cost of carbon (SCC) associated with electricity. [PV]² leverages the ability to use one BCN object for multiple alternatives.

Currently, the examples in this section do not address the following:

- Sub-year timesteps
 - Could be added to modified BLCC, EDGe\$, or [PV]² example files to test
- Other economic analysis types (e.g., profit maximization)
 - Could add a manufacturing example that includes revenue and profit terms

4.5.1 BLCC

The BLCC software provides comprehensive economic analysis of proposed capital investments that are expected to reduce long-term operating costs of buildings or building systems. Numerous alternative designs can be evaluated simultaneously to determine which has the lowest life-cycle cost. Comparative economic measures can be calculated for any design alternative relative to the designated base case. BLCC complies with ASTM building economics standards and federal requirements and guidance for both military and non-military projects. There are six different “modules” in BLCC targeting specific types of federal projects as described below.

FEMP Analysis, Energy Project: The criteria used as defaults in this module follow the life-cycle costing rules of the Federal Energy Management Program according to 10 Code of Federal Regulations (CFR) 436A as they apply to energy and water conservation and renewable energy projects funded by agencies from direct appropriations.

Federal Analysis, Financed Project: This is the module to be used for an LCC analysis of Energy Savings Performance Contracts (ESPC), Utility Energy Services Contracts (UESC) or other alternatively financed investments in energy or water conservation in the Federal Government. The criteria used as defaults in this module are applicable to all agencies in the Federal Government.

OMB Analysis, Non-Energy Project: This module supports analyses that are subject to the life-cycle costing guidelines of OMB Circular A-94 for the following types of projects:

- (a) Cost-effectiveness, lease-purchase, internal government investment, and asset sales
- (b) Public investment and regulatory analyses

MILCON Analysis, Energy Project: This module supports LCC analyses, according to 10 CFR 436A, of agency-funded energy and water and renewable energy projects for military construction in the Army, Navy, and Air Force.

MILCON Analysis, ECIP Project: This module can be used for generating the "ECIP Report" for Military Construction (MILCON) projects funded by the DoD Energy Conservation Investment Program (ECIP) to retrofit existing energy systems.

MILCON Analysis, Non-Energy Project: This module supports LCC analyses of new acquisition or construction projects, lease-purchase decisions, modification of existing facilities and similar projects the purpose of which is not primarily to assess

BLCC includes a file template as well as an example file for each of these project type modules. Each has its unique set of default assumptions based on the requirements and guidance for the given project type. The costs in BLCC are typically categorized as follows:

- Capital Component
 - Investment Cost
 - Replacement Costs
 - OM&R Costs – Annually Recurring
 - OM&R Costs – Non-Annually Recurring
- Contract Costs (*Financed Projects Only*)
 - Annually Recurring
 - Non-Annually Recurring
- Energy Costs
 - Electricity
 - Distillate Fuel Oil

- Residual Fuel Oil
- Natural Gas
- Liquefied Petroleum Gas
- Coal
- Water Costs

Several variations on this categorization exist depending on the analysis module. First, Contract Costs are only available in BLCC if the “Federal Analysis, Financed Project” type is selected. Costs related to contracts can be included in other modules by defining them as OM&R Costs. Second, the cost categorization is different for “MILCON Analysis, ECIP Project” because the baseline is assumed to have zero costs and each alternative defines its costs (or cost savings) relative to the baseline. The inputs for this project type are designed specifically for MILCON Energy Conservation Investment Program (ECIP), which is the predecessor to the Energy Resilience and Conservation Investment Program (ERCIP).

Four examples, each using a different project analysis module in BLCC, have been used to validate the calculations using both real and nominal dollars and discount rates, end-of-year and mid-year discounting conventions, varying service date relative to the base date, alternatives with and without different cost categorization types including initial investment costs, residual values, recurring and non-recurring future costs, contract costs, items with multiple cost aspects (i.e. electricity consumption costs and demand charges), items with a non-monetary value of interest (i.e. cost and quantity of electricity consumption), and alternatives with zero cost items.

FEMP Analysis, Energy Project: “FEMP Energy” includes initial investment costs, recurring future costs, non-recurring future costs, residual values, multiple cost aspects and non-monetary values for electricity, and uses real dollars and discount rates with end-of-year discounting convention. The service date is the same as the base date.

Federal Analysis, Financed Project: “Federal Financed” varies from “FEMP Energy” in that it does not include any investment costs, includes annual contract payments, and uses nominal dollars and discount rates with end-of-year discounting convention.

OMB Analysis, Non-Energy Project: “OMB Non-Energy” is used to validate using a different real discount rate and mid-year discounting convention.

MILCON Analysis, ECIP Project: “MILCON ECIP” is used to validate using the unique cost/savings input format for the module, alternative with zero cost items, and service date being different than the base date.

The assumptions for MILCON Analysis, Energy Project and MILCON Analysis, Non-Energy Project modules are covered within one or more of the other examples.

Note these examples do not validate the following:

- Multiple alternatives to the baseline

- Could use MILCON Non-Energy because it has three alternatives
- Covered by [PV]² if power purchase agreement/lease option is included in analysis
- Sub-year timesteps
- Continuous discounting
- Other economic analysis types (e.g., BCA, profit maximization, trade-offs)

4.5.1.1 End-of-Year Discounting

Two examples from BLCC use end-of-year discounting because both are non-military projects: FEMP Analysis – Energy and Federal Analysis – Financed.

4.5.1.1.1 FEMP Analysis - Energy Project

The example file for a FEMP Analysis, Energy Project named “FEMP Energy” (using 2021 data) has been used to generate an E3 input JSON string that was submitted manually to the E3 API. The criteria used as defaults in the FEMP Energy Analysis Project module follow FEMP’s life-cycle costing rules according to 10 CFR 436A as they apply to energy and water conservation and renewable energy projects funded by agencies from direct appropriations.

The project is “Heating/Cooling System” for a Park Service building in Washington, D.C. considering two alternatives: “Existing System” that keeps the existing baseboard heater and window air conditioning (AC) unit and “New System” that replaces the existing system with a heat pump. The analysis assumes a 15-year study period with base date of April 2021, end-of-year discounting, the DOE real discount rate (3 %), and provides all costs in constant/real dollar terms. The real DOE electricity price escalation rates for a commercial customer in Washington, D.C. shown in Table 4-28 are used for estimating electricity prices over the study period:

Table 4-28 Real Electricity Price Escalation Rates (Year-Over-Year Percentage Changes)

2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035
0.04	-1.00	-1.31	-0.95	-0.04	0.11	-0.04	0.23	-0.11	-0.46	-0.73	0.0	-0.12	-0.31	-0.54

This information maps to the E3 Analysis object as shown in Table 4-29:

Table 4-29 Analysis Object Values

Var Name	Var Type	Format/Values	Req
analysisType	String	LCCA	X
projType	String	Buildings	
objToReport	List of Strings	[FlowSummary, MeasureSummary]	X
studyPeriod	Int	15	X
baseDate	Date	2021-04-01	X
serviceDate	Date	2021-04-01	
timestepVal	String	Year	X
timestepComp	String	MidYear	X
outputRealBool	Boolean	True	X
interestRate	Float		
dRateReal	Float	0.03	X
dRateNom	Float		
inflationRate	Float		X
Marr	Float	0.03	X
reinvestRate	Float	0.03	X
incomeRateFed	Float		
incomeRateOther	Float		
noAlt	Int	2	X
baseAlt	Int	0	X
location	List of Strings	United States, DC, 20008	

The existing system includes the following costs:

- Capital Component: Baseboard Heater
 - Investment Costs
 - Initial Cost = \$500
 - Expected Life = 15 years
 - Residual Value Factor = 10 %
- Capital Component: Window AC Unit
 - Investment Costs
 - Initial Cost = \$1000
 - Expected Life = 15 years
 - Residual Value Factor = 10 %
 - Replacement Costs: Fan
 - Cost = \$600
 - Replacement/Expected Life = 10 years
 - Residual Value Factor = 50 % ()
 - OM&R Costs – Annually Recurring
 - Routine Maintenance
 - \$50
 - Annual Rate of Increase = 0 %
 - OM&R Costs – Non-Annually Recurring
 - Cleaning

- \$200
 - Occurrence = 10 years
 - Annual Rate of Increase = 0 %
- Energy Costs
 - Electricity
 - 15 000 kWh
 - \$0.08/kWh
 - DOE real electricity price escalation rates
 - No demand charge or annual utility rebates

These costs map to E3 BCN objects as shown in Table 4-30 for the existing system.

Table 4-30 BCN Object Values – Existing System

VarName	Baseboard Heater	AC Unit	Fan Replace.	Electricity	Maint.	Cleaning	BB RV	AC RV	Fan RV
altID	0	0	0	0	0	0	0	0	0
bcnID	0	1	2	3	4	5	6	7	8
bcnType	Cost	Cost	Cost	Cost	Cost	Cost	Cost	Cost	Cost
bcnSubType	Direct	Direct	Direct	Direct	Direct	Direct	Direct	Direct	Direct
bcnName	Baseboard Heater	AC Unit	Fan Replacement	Electricity	Maintenance	Cleaning	BB Residual Value	AC Residual Value	Fan Residual Value
bcnTag	Initial Investment	Initial Investment	Replacement	Energy	OMR	OMR	Salvage Value	Salvage Value	Salvage Value
initialOcc	0	0	12	1	1	10	15	15	15
bcnRealBool	true	true	true	true	true	true	true	true	true
bcnInvestBool	true	true	true	false	false	False	true	true	true
bcnLife	15	15	12	Null	Null	Null	Null	Null	Null
rvBool	false	false	false	false	false	False	true	true	true
rvOnly	false	false	false	false	false	false	true	true	True
recurBool	false	false	true	true	true	True	false	false	false
recurInterval	Null	Null	12	1	1	10	Null	Null	Null
recurVarRate	Null	Null	Null	Percent Delta Timestep X-1	null	null	Null	Null	Null
recurVarValue	Null	Null	Null	[DOE Esc Rate List]	null	null	Null	Null	Null
recurEndDate	Null	Null	15	15	15	Null	Null	Null	Null
valuePerQ	500	1000	600	0.08	50	200	-50	-100	-300
quant	1	1	1	15000	1	1	1	1	1
quantVarRate	Null	Null	Null	Fraction Delta Timestep X-1	Null	Null	Null	Null	Null
quantVarValue	Null	Null	Null	[0.0]	Null	Null	Null	Null	Null
quantUnit	Null	Null	Null	kWh	Null	Null	Null	Null	Null

The new system includes the following costs:

- Capital Component: Heat Pump
 - Investment Costs
 - Initial Cost = \$3000
 - Expected Life = 20 years
 - Residual Value Factor = 25 %

- OM&R Costs – Annually Recurring
 - Routine OM
 - \$100
 - Annual Rate of Increase = 0 %
- OM&R Costs – Non-Annually Recurring
 - Overhaul
 - \$600
 - Annual Rate of Increase = 0 %
- Energy Costs
 - Electricity
 - 10 250 kWh
 - \$0.08/kWh
 - DOE electricity price escalation rates
 - No demand charge or annual utility rebates

These costs map to E3 BCN objects as shown in Table 4-31 for the new system.

Table 4-31 BCN Object Values – New System

VarName	Heat Pump	Electricity	Routine OM	Overhaul	Heat Pump RV
altID	1	1	1	1	1
bcnID	8	9	10	11	12
bcnType	Cost	Cost	Cost	Cost	Cost
bcnSubType	Direct	Direct	Direct	Direct	Direct
bcnName	Heat Pump	Electricity	Maintenance	Cleaning	Heat Pump Residual Value
bcnTag	Initial Investment	Energy	OMR	OMR	Salvage Value
initialOcc	0	1	1	10	15
bcnRealBool	true	true	true	true	true
bcnInvestBool	1	0	0	0	1
bcnLife	20	Null	Null	Null	Null
rvBool	false	false	false	false	true
rvOnly	false	false	false	false	true
recurBool	true	true	true	true	false
recurInterval	Null	1	1	8	Null
recurVarRate	Null	Percent Delta Timestep X-1	Null	Null	Null
recurVarValue	Null	[DOE Esc Rate List]	Null	Null	Null
recurEndDate	Null	20	20	Null	Null
valuePerQ	3000	0.08	100	600	-750
quant	1	10250	1	1	1
quantVarRate	Null	Null	Null	Null	Null
quantVarValue	Null	Null	Null	Null	Null
quantUnit		kWh			

The cash flow results from the E3 API reported in “FlowSummary” – “TotCostNonDisc” were compared to the results in the BLCC reports in Table 4-32 to confirm consistency between E3 and BLCC. Within \$1 for every year’s cash flow for both alternatives.

Table 4-32 Cash Flow Comparison

Results Comparison	Nominal Cash Flow (\$)			
	Existing System		New System	
Year	BLCC	E3	BLCC	E3
0	\$1500*	\$1500	3000*	\$3000
1	\$1250*	\$1250	\$920*	\$920
2	\$1239	\$1238	\$912	\$912
3	\$1223	\$1223	\$902	\$901
4	\$1212	\$1212	\$894	\$894
5	\$1211	\$1211	\$894	\$894
6	\$1213	\$1213	\$895	\$894
7	\$1212	\$1212	\$894	\$894
8	\$1215	\$1215	\$896	\$896
9	\$1214	\$1214	\$1495	\$1495
10	\$1208	\$1208	\$892	\$891
11	\$2000	\$2000	\$886	\$886
12	\$1200	\$1200	\$886	\$886
13	\$1199	\$1198	\$885	\$885
14	\$1195	\$1195	\$882	\$882
15	\$739	\$739	\$128	\$128
BLCC reports Year 0 and Year 1 cash flows together. Here the results are broken down to how they are discounted.				

The difference in the LCC results as shown in Table 4-33 is minimal ($\approx 0.1\%$) and a result of rounding during the discounting and escalation calculations.

Table 4-33 Summary Results Comparison – FEMP Energy Project

Results Comparison	Existing System		New System	
Measure	BLCC	E3	BLCC	E3
Total Present Value Life Cycle Costs	\$16 297	\$16 277	\$13 680	\$13 665
Net Savings			\$2617	\$2613
SIR Ratio			4.04	3.99
AIRR			13.05 %	12.95 %
SPP			5	5
DPP			6	6
Annual Energy Consumption (kWh)	15 000	15 000	10 250	10 250
Total Energy Consumption (kWh)	225 000	225 000	153 750	153 750
Total Energy Reduction (kWh)			71 240	71 250
BLCC Service Date = 04/2021, but annual timestep assumes 15 complete years BLCC reports Year 0 and Year 1 results together, but does not discount Year 0 costs Differences in Total LCC is due to rounding during discounting and escalation calculations. Cash Flows are identical				

Note: The results do not currently compare emissions reduction estimates. This could be completed by finding the source file within BLCC to obtain the emissions factors for each fuel type for Washington, DC and then adding non-monetary BCN objects with the same information as the electricity BCN objects except replacing the quantity and units. An example of this is available from the [PV]² example.

4.5.1.1.2 Federal Analysis – Financed Project

The example file for a FEMP Analysis, Financed Project named “Federal Financed” (using 2021 data) has been used to generate an E3 input JSON string that was submitted manually to the E3 API. The criteria used as defaults in the FEMP Energy Analysis Project module follow FEMP’s life-cycle costing rules according to 10 CFR 436A as they apply to energy and water conservation and renewable energy projects funded through financing instead of the previous example that used direct appropriations. As a result, the analysis is completed in current/nominal terms and requires some adjustments to the costs data to function in E3 that are discussed below.

The project is “Lighting/Daylighting” for a federal building in Arizona considering two alternatives: “Existing” that keeps the existing lighting system and “Lighting Retrofit” that retrofits the lighting system to reduce energy consumption. The analysis assumes a 15-year study period (expected remaining life on existing lighting system) with base date of April 2021, end-of-year discounting, the DOE nominal discount rate (1.5 %) and implied inflation rate (-1.5 %)⁵, and provides all costs in current/nominal dollar terms. The nominal DOE electricity price escalation rates for a commercial customer in Arizona shown in Table 4-34 are used for estimating electricity prices and demand charges over the study period. Note that the nominal escalation rate estimates are negative because electricity prices are projected to increase at a slower rate than general inflation (or even decrease in some years).

Table 4-34 Nominal Electricity Price Escalation Rates (Year-Over-Year Percentage Changes)

2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035
-2.22	-2.79	-2.97	-0.87	-0.27	-1.39	-1.91	-1.85	-1.17	-0.74	-1.45	-1.69	-1.58	-1.58	-1.82

This information maps to the E3 Analysis object as shown in Table 4-35:

⁵ The implied inflation rate is a result of an inconsistency created from the 3.0 % floor on real discount rates defined in the CFR. The nominal escalation rate is calculated using nominal yields on long-term treasuries, resulting in a nominal discount rate below the floor (1.5 % versus 3.0 %). The inflation rate that will make the nominal and real discount rates consistent is -1.5 %, which is different than the inflation rate used by the Council of Economic Advisors (2.3 %) that is defined to be used in the CFR.

Table 4-35 Analysis Object Values

Var Name	Var Type	Format/Values	Req
analysisType	String	LCCA	X
projType	String	Buildings	
objToReport	List of Strings	[FlowSummary, MeasureSummary, OptionalSummary]	X
studyPeriod	Int	15	X
baseDate	Date	2021, 04, 01	X
serviceDate	Date	2021, 04, 01	
timestepVal	String	Year	X
timestepComp	String	EndOfYear	X
outputRealBool	Boolean	false	X
interestRate	Float		
dRateReal	Float		
dRateNom	Float	0.015	X
inflationRate	Float	-0.015	X
Marr	Float	0.03	X
reinvestRate	Float	0.03	X
incomeRateFed	Float		
incomeRateOther	Float		
noAlt	Int	2	X
baseAlt	Int	0	X
location	List of Strings	United States, AZ, 85364	

The existing system includes the following costs:

- Capital Component: Existing System
 - Investment Costs
 - \$0
 - OM&R Costs – Annually Recurring
 - \$5600
 - Annual Rate of Change = -1.5 %
 - Value is in Year 0 dollars and be converted to current/nominal dollars for every future year
 - Assumed to change at the general rate of inflation (-1.5 %) to make the real present value the same in each year
- Energy Costs
 - Electricity
 - 1 082 633 kWh
 - \$0.046/kWh
 - DOE nominal electricity price escalation rates
 - \$30 105 demand charge

These costs map to E3 BCN objects as shown in Table 4-36:

Table 4-36 BCN Object Values – Existing System

VarName	Existing System	Electricity - Consumption	Electricity – Demand Charge	OMR Cost
altID	0	1	2	3
bcnID	0	3		4
bcnType	Cost	Cost		Cost
bcnSubType	Direct	Direct		Direct
bcnName	Existing System	Electricity-Consumption	Electricity-Demand Charge	OMR Cost
bcnTag	Initial Investment	Energy-Consumption	Energy-Demand Charge	OMR
initialOcc	0	1	1	1
bcnRealBool	false	false	false	false
bcnInvestBool	true	false	False	false
bcnLife	15	Null	null	Null
rvBool	False	false	false	false
rvOnly	false	false	false	false
recurBool	Null	True	true	true
recurInterval	Null	1	1	1
recurVarRate	Percent Delta Timestep X-1	Percent Delta Timestep X-1	Percent Delta Timestep X-1	Percent Delta Timestep X-1
recurVarValue	-0.015	[DOE Esc Rate List]	[DOE Esc Rate List]	-0.015
recurEndDate	Null	Null	Null	Null
valuePerQ	0	0.046	30105	5600
quant	1	1082633	1	1
quantVarRate	Null	Null	Null	Null
quantVarValue	Null	Null	Null	Null
quantUnit	Null	kWh	null	Null
The Existing System object was included in the example for completeness to match the BLCC example file and could be excluded from the JSON file submission because it has no value.				

The new system includes the following costs:

- Capital Component: New System
 - Investment Costs
 - Initial Cost Paid by Agency = \$0
 - Initial Cost Financed = \$390 480
 - Covered in Contract Cost objects
 - Expected Life = 20 years
 - Annual Rate of Increase = -1.50 %
 - Assumed to increase at the general rate of inflation
 - Residual Value Factor = 25 %
 - Calculated based on the adjusted value at the end of the study period (Year 15)
 - \$390 480 adjusted by -1.50 % annually
 - OM&R Costs – Annually Recurring
 - Post-Contract Operation and Maintenance (OM) Costs
 - \$3000
 - Annual Rate of Increase = -1.5 %
 - Assumed to increase at the general rate of inflation
 - Year 11 through Year 15

- Energy Costs
 - Electricity
 - 206 911 kWh
 - \$0.046/kWh
 - DOE nominal electricity price escalation rates
 - \$3311 demand charge
- Contract Costs – Annually Recurring
 - Annual Contract Payment
 - \$67000
 - -1.60 % escalation rate
 - Usage Indices
 - Set to 100.0 % for the length of the contract and 0.0 % for remainder
 - Usage Factor 100.0 % for 10 years
 - Usage Factor 0.0 % for remaining (Year 11 through Year 15)

These costs map to E3 BCN objects as shown in Table 4-37.

Table 4-37 BCN Object Values – New System

[illegible]

The cash flow results from the E3 API reported in “FlowSummary” – “TotCostNonDisc” were compared to the results in the BLCC reports in Table 4-38 to confirm consistency between E3 and BLCC. Within \$14 (0.02 %) for every year’s cash flow for both alternatives.

Table 4-38 Cash Flow Comparison

Results Comparison	Nominal Cash Flow (\$)			
	Existing System		New System	
Year	BLCC	E3	BLCC	E3
0	0	0	0	0
1	\$83 656	\$83 648	\$78 478	\$78 472
2	\$81 400	\$81 386	\$77 076	\$77 067
3	\$79 061	\$79 048	\$75 674	\$75 667
4	\$78 338	\$78 327	\$74 552	\$74 543
5	\$78 062	\$78 051	\$73 516	\$73 506
6	\$76 973	\$76 960	\$72 367	\$72 355
7	\$75 525	\$75 511	\$71 172	\$71 161
8	\$74 145	\$74 132	\$70 007	\$69 994
9	\$73 258	\$73 248	\$68 935	\$68 922
10	\$72 678	\$72 669	\$67 928	\$67 914
11	\$71 623	\$71 613	\$13 278	\$13 277
12	\$70 424	\$70 411	\$13 059	\$13 057
13	\$69 315	\$69 303	\$12 855	\$12 853
14	\$68 224	\$68 211	\$12 654	\$12 652
15	\$66 996	\$66 984	-\$65 390	-\$65 389
BLCC reports Year 0 and Year 1 cash flows together. Here the results are broken down to how they are discounted.				

The difference in the results is minimal ($\approx 0.02\%$) as shown in Table 4-39 and a result of rounding during the discounting and escalation calculations.

Table 4-39 Summary Results Comparison – Federal Financed Energy Project

Results Comparison	Existing System		New System	
	BLCC	E3	BLCC	E3
Total Present Value Life Cycle Costs	\$1 000 117	\$ 999 909	\$665 055	\$664 920
Net Savings			\$335 062	\$ 334 988
Annual Energy Consumption (kWh)	1 082 633	1 082 633	206 911	206 911
Total Energy Consumption (kWh)	16 239 495	16 239 495	3 103 665	3 103 665
Total Energy Reduction (kWh)			13 134 032	13 135 830
BLCC Service Date = 04/2021, but annual timestep assumes 15 complete years				
BLCC reports Year 0 and Year 1 results together, but does not discount Year 0 costs				
Differences in Total LCC is due to rounding during discounting and escalation calculations.				
No initial investment costs leads to the inability to calculate the SIR, AIRR, SPP, and DPP.				

Note: The results do not currently compare emissions reduction estimates. This could be completed by finding the source file within BLCC to obtain the emissions factors for

each fuel type for Washington, DC and then adding non-monetary BCN objects with the same information as the electricity BCN objects except replacing the quantity and units. An example of this is available from the [PV]² example.

4.5.1.2 Mid-Year Discounting

Two examples from BLCC use mid-year discounting because both are military projects: OMB Analysis – Non-Energy and MILCON Analysis ECIP.

4.5.1.2.1 OMB Analysis – Non-Energy Project

The example file for an Office of Management and Budget (OMB) Analysis, Non-Energy Project named “OMB Non-Energy” (using 2021 data) has been used to generate an E3 input JSON string that was submitted manually to the E3 API. The criteria used as defaults in the OMB Non-Energy Analysis Project module follow OMB’s life cycle costing rules according to 10 CFR 436A as they apply to non-energy related federal projects.

The project is “OMB Demo” for a buy versus lease decision for a building in Maryland. Therefore, the analysis purpose is “cost-effectiveness, lease purchase, internal government investment or asset sale analysis.” The baseline is “Lease” and the alternative is “Buy.” The analysis assumes a 15-year study period with base date of April 2021, mid-year discounting (presumably a DoD project), the real OMB discount rate for projects of 15 years (-0.8 %; average of 10-year and 20-year discount rates from Circular A-94), and provides all costs in constant/real dollar terms. This information maps to the E3 Analysis object as shown in Table 4-40:

Table 4-40 Analysis Object Values

Var Name	Var Type	Format/Values	Req
analysisType	String	LCCA	X
projType	String	Buildings	
objToReport	List of Strings	[FlowSummary, MeasureSummary, OptionalSummary]	X
studyPeriod	Int	15	X
baseDate	Date	2021, 04, 01	X
serviceDate	Date	2021, 04, 01	
timestepVal	String	Year	X
timestepComp	String	MidYear	X
outputRealBool	Boolean	true	X
interestRate	Float		
dRateReal	Float	-0.008	X
dRateNom	Float		
inflationRate	Float		
Marr	Float	-0.008	X
reinvestRate	Float	-0.008	X
incomeRateFed	Float		
incomeRateOther	Float		
noAlt	Int	2	X
baseAlt	Int	0	X
location	List of Strings	United States, MD	

The example assumes that operational costs are identical between the two examples. The only difference is the purchase option includes the initial purchase cost, residual value, and maintenance costs while the leasing option only includes the lease payments.

The baseline (lease) alternative includes two cost objects:

- Capital Component: Leasing
 - Investment Costs
 - Initial Cost = \$0
 - Expected Life = 50 years
 - Annual Rate of Increase = 0 %
 - Residual Value Factor = 0 %
 - OM&R Costs – Annually Recurring
 - Lease Payment
 - \$500 000
 - Annual Rate of Increase = 0 %

The buy alternative includes two cost objects:

- Capital Component: Buy
 - Investment Costs
 - Initial Cost = \$5 000 000

- Expected Life = 50 years
- Annual Rate of Increase = 0 %
- Residual Value Factor = 50 %
- OM&R Costs – Annually Recurring: Building Maintenance
 - \$200 000
 - Annual Rate of Increase = 0 %

These costs map to E3 BCN objects as shown in Table 4-41:

Table 4-41 BCN Object Values

VarName	Leasing		Buying		
	Initial Cost	Lease Payment	Buy Cost	Building Maintenance	Residual Value
altID	0	1	2	3	4
bcnID	0	0	1	1	1
bcnType	Cost	Cost	Cost	Cost	Cost
bcnSubType	Direct	Direct	Direct	Direct	Direct
bcnName	Initial Cost	Lighting Retrofit Contract Payment	Initial Cost	Building Maintenance	Residual Value
bcnTag	Initial Investment	OMR	Initial Investment	OMR	Salvage Value
initialOcc	0	1	0	1	15
bcnRealBool	true	true	true	true	true
bcnInvestBool	True	False	True	false	True
bcnLife	50	Null	50	Null	Null
rvBool	False	false	False	false	true
rvOnly	false	false	false	false	True
recurBool	Null	True	Null	true	false
recurInterval	Null	1	Null	1	Null
recurVarRate	null	null	null	null	Null
recurVarValue	Null	Null	Null	Null	Null
recurEndDate	Null	Null	Null	Null	Null
valuePerQ	0	500000	5000000	200000	-2500000
quant	1	1	1	1	1
quantVarRate	Null	Null	Null	Null	Null
quantVarValue	Null	null	Null	Null	Null
quantUnit	Null	Null	Null	Null	Null
Initial Cost object for the lease option is included for completeness for the BLCC example and could be excluded from the E3 JSON submission.					

The cash flow results from the E3 API reported in “FlowSummary” – “TotCostNonDisc” were compared to the results in the BLCC reports in Table 4-42 to confirm consistency between E3 and BLCC with a perfect match for all values.

Table 4-42 Cash Flow Comparison

Results Comparison	Nominal Cash Flow (\$)			
	Lease		Buy	
Year	BLCC	E3	BLCC	E3
0	\$0	\$0	\$5 000 000*	\$5 000 000*
1	\$500 000	\$500 000	\$200 000*	\$200 000*
2	\$500 000	\$500 000	\$200 000	\$200 000
3	\$500 000	\$500 000	\$200 000	\$200 000
4	\$500 000	\$500 000	\$200 000	\$200 000
5	\$500 000	\$500 000	\$200 000	\$200 000
6	\$500 000	\$500 000	\$200 000	\$200 000
7	\$500 000	\$500 000	\$200 000	\$200 000
8	\$500 000	\$500 000	\$200 000	\$200 000
9	\$500 000	\$500 000	\$200 000	\$200 000
10	\$500 000	\$500 000	\$200 000	\$200 000
11	\$500 000	\$500 000	\$200 000	\$200 000
12	\$500 000	\$500 000	\$200 000	\$200 000
13	\$500 000	\$500 000	\$200 000	\$200 000
14	\$500 000	\$500 000	\$200 000	\$200 000
15	\$500 000	\$500 000	-\$2 300 000	-\$2 300 000
BLCC reports Year 0 and Year 1 cash flows together. Here the results are broken down to how they are discounted.				

The difference in the results as shown in Table 4-43 is minimal (0.43 %) and a result of rounding during the discounting and escalation calculations.

Table 4-43 Summary Results Comparison – OMB Non-Energy Project

Results Comparison	Lease		Buy	
Measure	BLCC	E3	BLCC	E3
Total Present Value Life Cycle Costs	\$ 7 970 462	\$7 970 493	\$ 5 368 129	\$5 379 398
Net Savings			\$ 2 602 334	\$ 2 591 096
SIR Ratio			2.19	2.18
AIRR			4.53 %	4.50 %
SPP			15	15
DPP			15	15
BLCC Service Date = 04/2021, but annual timestep assumes 15 complete years BLCC reports Year 0 and Year 1 results together, but does not discount Year 0 costs Differences in Total LCC is due to rounding during discounting and escalation calculations. Currently E3 does not allow mid-point discounting, leading to differences.				

Note: The results do not currently compare emissions reduction estimates. This could be completed by finding the source file within BLCC to obtain the emissions factors for each fuel type for Washington, DC and then adding non-monetary BCN objects with the same information as the electricity BCN objects except replacing the quantity and units. An example of this is available from the [PV]² example.

4.5.1.2.2 MILCON Analysis - ECIP Project

The example file for a MILCON Analysis, ECIP Project named “MILCON ECIP” (using 2021 data) has been used to generate an E3 input JSON string that was submitted manually to the E3 API. The criteria used as defaults in the MILCON ECIP Analysis Project module follow FEMP’s life-cycle costing rules according to 10 CFR 436A as they apply to energy and water conservation and renewable energy projects funded by agencies from direct appropriations. This module is structured differently than the other BLCC modules because it requires inputting alternative information as cost savings related to the baseline. Therefore, no inputs are required for the baseline.

The project is “Install DX Split System AC” in family housing office at Dahlgren, VA, Naval Station. The baseline is keeping the existing window air conditioners and assumes zero costs or cost savings. The alternative being considered is replacing the existing window air conditioners with a direct expansion (DX) split system. The analysis assumes a 21-year study period with base date of April 2021, “beneficial occupancy date” of 1 year after the base date, mid-year discounting, the DOE real discount rate (3 %), and provides all costs in constant/real dollar terms. The real DOE electricity price escalation rates for an industrial customer in Virginia as shown in Table 4-44 are used for estimating electricity prices over the study period:

Table 4-44 Real Electricity Price Escalation Rates (Year-Over-Year Percentage Changes)

2021	2022	2023	2024	2025	2026	2027	2028	2029	2030
0.39	-1.93	-2.2	-0.98	0.52	0.29	0.23	0.29	-0.06	-0.23

2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041
-0.35	-0.12	-0.12	-0.17	-0.52	-0.52	-0.29	-0.53	-0.47	-0.42	-0.54

This is the first example that includes a delay between the base date and occupancy date.

This information maps to the E3 Analysis object as shown in Table 4-45:

Table 4-45 Analysis Object Values

Var Name	Var Type	Format/Values	Req
analysisType	String	LCCA	X
projType	String	Buildings	
objToReport	List of Strings	[FlowSummary, MeasureSummary]	X
studyPeriod	Int	15	X
baseDate	Date	2021, 04, 01	X
serviceDate	Date	2022, 04, 01	
timestepVal	String	Year	X
timestepComp	String	MidYear	X
outputRealBool	Boolean	true	X
interestRate	Float		
dRateReal	Float	0.03	X
dRateNom	Float		
inflationRate	Float	-0.015	
Marr	Float	0.03	X
reinvestRate	Float	0.03	X
incomeRateFed	Float		
incomeRateOther	Float		
noAlt	Int	2	X
baseAlt	Int	0	X
location	List of Strings	United States, VA	

The alternative includes the following additional costs and cost savings:

- Capital Component Savings/Cost
 - Additional Investment Costs
 - Construction Cost = \$142 800
 - SIOH = \$10 200
 - Design Cost = \$17 000
 - Salvage Value of Existing Equipment = \$0
 - Public Utility Company Rebate = \$0
 - Annually Recurring Savings
 - Routine OM&R
 - Amount Saved = \$2270
 - Annual Rate of Increase = 2.0 %
 - Non-Annually Recurring Savings
 - Major Repair1 (Avoided Repair)
 - Amount Saved = \$9000
 - Occurrence = 6 years
 - Annual Rate of Increase = 0 %
 - Major Repair2 & Replacement
 - Amount Saved = -\$9000
 - Occurrence = 12 years

- Annual Rate of Increase = 0 %
- Major Repair 3
 - Amount Saved = \$12 000
 - Occurrence = 18 years
 - Annual Rate of Increase = 0 %
- Scheduled Repair1
 - Amount Saved = -\$1500
 - Occurrence = 5 years
 - Annual Rate of Increase = 0 %
- Scheduled Repair2
 - Amount Saved = -\$1500
 - Occurrence = 10 years
 - Annual Rate of Increase = 0 %
- Scheduled Repair3
 - Amount Saved = -\$1500
 - Occurrence = 15 years
 - Annual Rate of Increase = 0 %
- Energy Costs
 - Electricity
 - 215 300 kWh
 - \$0.08711/kWh
 - DOE real electricity price escalation rates for industrial sector
 - No demand charge or annual utility rebates

Note that there are both positive and negative savings values because there are both avoided costs (positive savings) as well as additional costs (negative savings) associated with the project relative to the baseline.

These costs map to E3 BCN objects as shown in Table 4-46.

Table 4-46 BCN Object Values

VarName	Additional Investment Cost	Routine OM&R	Major Repair1	Major Repair2 & Replacement	Major Repair3	Scheduled Repairs	Electricity	Zero Baseline Cost
altID	1	1	1	1	1	1	1	0
bcnID	0	1	2	3	4	5	6	7
bcnType	Cost	Cost	Cost	Cost	Cost	Cost	Cost	Cost
bcnSubType	Direct	Direct	Direct	Direct	Direct	Direct	Direct	Direct
bcnName	Additional Investment Cost	Routine OM&R	Major Repair1	Major Repair2 & Replacement	Major Repair3	Schedule Repairs	Electricity	Zero Baseline Cost
bcnTag	Initial Investment	Routine OMR	Major Repair1	Major Repair2	Major Repair3	Scheduled Repairs	Energy	Initial Investment
initialOcc	0	2	7	13	19	6	2	0
bcnRealBool	true	true	true	true	true	true	true	true
bcnInvestBool	true	false	false	False	false	False	false	True
bcnLife	null	null	Null	Null	Null	Null	Null	Null
rvBool	false	false	false	false	false	false	false	False
rvOnly	false	false	false	false	false	False	false	false
recurBool	false	false	False	False	False	True	true	false
recurInterval	Null	Null	Null	Null	Null	5	1	Null
recurVarRate	Null	Percent Delta Timestep X-1	Null	Null	Null	Null	Percent Delta Timestep X-1	Null
recurVarValue	Null	0.02	Null	Null	Null	Null	[DOE Esc Rate List]	Null
recurEndDate	Null	Null	Null	Null	Null	20	Null	Null
valuePerQ	170000	-2270	-9000	9000	-12000	1500	0.08711	0
quant	1	1	1	1	1	1	-215300	1
quantVarRate	Null	Null	Null	Null	Null	Null	null	Null
quantVarValue	Null	Null	Null	Null	Null	Null	Null	Null
quantUnit	Null	Null	Null	Null	Null	Null	kWh	Null
Total Investment is used for Additional Investment Costs. Each cost line item could be included individually for greater results details. Scheduled Repairs are combined into a single recurring object. Each repair could be included individually if desired. Each alternative requires at least one BCN object. Therefore, Zero Baseline Cost is included.								

The occurrence dates in BLCC are based on initial occupancy (i.e., service date), which is one year after the base date. The scheduled repair does not occur in year 21 because the system is assumed to reach the end of its service life. The cost savings are included as negative costs in the BCN objects.

The results by line item from the E3 API were compared to the results in the BLCC reports in Table 4-47 to confirm consistency between E3 and BLCC. The ECIP Report in BLCC does not provide annual cash flows. Instead, it provides the discounted savings for each cost/savings item, which are all within 1.5 % of the BLCC results.

Table 4-47 Discounted Savings Comparison

Results Comparison	Discounted Savings by Item (\$)	
Item	BLCC	E3
Total Investment	-\$170 000	-\$170 000
Energy and Water Savings	\$263 407	\$262 823
Routine OM&R	\$40 845	\$41 251
Major Repair1	\$7318	\$7427
Major Repair2 & Replacement	-\$6129	-\$6220
Major Repair3	\$6843	\$6945
Scheduled Repairs	-\$3275	-\$3323
Total Savings	\$139 010	\$138 903
BLCC reports Year 0 and Year 1 cash flows together. Here the results are broken down to how they are discounted.		

The difference in the results is minimal (0.08 %) as shown in Table 4-48 and a result of rounding during the discounting and escalation calculations.

Table 4-48 Summary Results Comparison – MILCON ECIP Project

Results Comparison	System Replacement	
Measure	BLCC	E3
Net Savings	\$139 010	\$138 903
SIR Ratio	1.82	1.82
AIRR	5.97 %	5.97 %
SPP	7.95	9
Differences in Total LCC is due to rounding during discounting and escalation calculations. DPP not reported in BLCC.		

Note: The results do not currently compare emissions reduction estimates. This could be completed by finding the source file within BLCC to obtain the emissions factors for each fuel type for Washington, DC and then adding non-monetary BCN objects with the same information as the electricity BCN objects except replacing the quantity and units. An example of this is available from the [PV]² example.

4.5.2 Economic Decision Guide Software (EDGE\$)

EDGE\$ – Economic Decision Guide Software – provides a standard economic methodology for evaluating investment decisions required to improve the ability of communities to adapt to, withstand, and quickly recover from natural, technology, and human-caused disruptive events. The tool helps the user to identify and compare the relevant present and future resilience costs and benefits associated with new capital investment versus maintaining a community's status-quo. Benefits include cost savings and damage loss avoidance because enhancing resilience on a community scale creates value, including co-benefits, even if a hazard event does not strike.

EDGE\$ is based on the process found in NIST's Community Resilience Economic Decision Guide for Buildings and Infrastructure Systems (EDG). The EDG and EDGE\$

can be used as standalone tools, but are designed as part of a more comprehensive planning process and in combination with the NIST Community Resilience Planning Guide for Buildings and Infrastructure Systems.

This tool has allowed for validation of the benefit-cost analysis calculations and all related economic measures/metrics in E3 (i.e., NPV cash flows for benefits and costs, net benefits (NB), and benefit-cost ratio (BCR)). Additionally, EDGe\$ allows validation of E3's continuous discounting methodology. Instead of calculating and adding to the principal at specific intervals as done in discrete discounting, continuous discounting uses a natural log-based formula to calculate and add back accrued interest at the smallest possible intervals. For additional details, please visit <https://www.nist.gov/services-resources/software/edge-economic-decision-guide-software-online-tool> to access both the EDG (<http://dx.doi.org/10.6028/NIST.SP.1197>) and EDGe\$ (<https://edges.nist.gov/>).

Three BCA examples were submitted manually to the E3 API to validate the BCA calculations for continuous discounting:

- Results comparisons to the EDGe\$ Web interface using continuous discounting
- Results comparisons to the EDGe\$ Web interface using continuous discounting with sensitivity and uncertainty estimates

The key assumption within the EDGe\$ software is that all costs are treated as investment costs. E3 provides the flexibility to define non-investment costs. For the examples below, all costs are defined as investment costs within the E3 calculations to provide a direct comparison to the results from EDGe\$.

4.5.2.1 Continuous Discounting

Examples are based on the BCA of different real world natural disasters that have been evaluated by AEO using the Economic Decision Guide and EDGe\$. The examples from EDGe\$ required modification because EDGe\$ uses continuous discounting, and therefore required modification to be complete as a deterministic analysis with discrete discounting. The examples were recreated and modified in Excel to provide the base results for which to compare and validate the BCA results generated by E3.

4.5.2.1.1 Hurricane Resilience

This example is based on the Buyout/Levee Case Study in Appendix B.2 of the Economic Decision Guide, which assumes that after a recent major flood, a city decides to adopt a flood mitigation strategy to reduce expected impacts of potential future flooding. Two mutually exclusive alternatives are considered: (1) buyout all properties in the 100-year floodplain and turn the area into greenspace or (2) build a 1.5-mile-long levee designed for a 100-year flood.

Assumptions for the benefit-cost analysis are as follows:

- Planning Horizon: 75 years
- Real Discount Rate: 5.00%
- Interest Rate: 5.00%
- Inflation Rate: 2.00%
- MARR: 4.00%
- Reinvestment Rate: 5.00 %
- Disaster Rate: Every 100 years
- Disaster Magnitude: 0 % of build cost
- Statistical Value of a Life: \$750 000⁶

This information maps to the E3 Analysis object as shown in Table 4-49:

Table 4-49 Analysis Object Values

Var Name	Var Type	Format/Values	Req
analysisType	String	BCA	X
projType	String	Infrastructure	
objToReport	List of Strings	[FlowSummary, MeasureSummary]	X
studyPeriod	Int	75	X
baseDate	Date	2012, 04, 23	X
serviceDate	Date	2013, 04, 23	
timestepVal	String	Year	X
timestepComp	String	Continuous	X
outputRealBool	Boolean	True	X
interestRate	Float	Null	
dRateReal	Float	0.05	X
dRateNom	Float		
inflationRate	Float	0.02	X
Marr	Float	0.04	X
reinvestRate	Float	0.05	X
incomeRateFed	Float		
incomeRateOther	Float		
noAlt	Int	2	X
baseAlt	Int	0	X
location	List of Strings	United States, MD, 20879	

The mitigation measures are focused on a residential area with a high-risk of flooding within the 100-year flood plain. The area consists of 600 homes in total (average value of \$130 000), with 100 being considered waterfront properties (average value \$190 000). The buyout option would require all homes to be purchased, while the levee would require all 100 waterfront properties, and 100 additional properties, be purchased to make room for levee construction. Two hundred of the homes, including all homes in the levee construction area, are eligible for FEMA grants, which cover 75 % of the cost. The tax

⁶ Note that the statistical value of a life used in this example is different than in the other BCA examples and is for demonstration purposes only.

rate on all properties is 1.52 %. Regardless of the option, it is assumed that 25 % of the purchased homes do not return to the city's tax base.

Based on the benefit and cost information above, the benefits and costs of each alternative are shown in Table 4-50. Costs include initial investment costs associated with purchase and demolition of homes, direct and indirect design and construction costs of the mitigation strategy, and future OMR costs that are treated future investment costs with zero residual value (for consistency with EDGe\$ assumptions). Disaster related benefits include response and recovery costs, direct and indirect loss reductions, and reduced fatalities. Externalities can be both positive (benefits) and negative (costs). For this example, all externalities are treated as external benefits with externality costs defined as a negative external benefit to align with EDGe\$. The buyout leads to a large decrease in tax revenue while the levee leads to a smaller decrease in tax revenue while also flooding downstream of the levee, all of which are negative external benefits. Positive external benefits are only expected for the buyout option, including better stormwater management, value of greenspace, and increased visitation/tourism. Note that the negative external benefits could be inputted as positive external costs and the results would remain the same.

Table 4-50 Benefits and Costs by Alternative

Desc.	Tag	Object Name	Recurr.	Sub-Type	Type	Buyout	Levee
Costs	Initial Investment	Purchase of Homes	One-time	Direct	Cost	\$59 897 500	\$31 590 000
	Initial Investment	Demolition of Homes	One-time	Direct	Cost	\$5 400 000	\$2 400 000
	Initial Investment	Greenspace Const.	One-time	Direct	Cost	\$200 000	
	Initial Investment	Levee Design	One-time	Direct	Cost		\$2 250 000
	Initial Investment	Levee Construction	One-time	Direct	Cost		\$15 000 000
	Initial Investment	Indirect Costs	One-time	Indirect	Cost	\$1 680 000	\$4 500 000
	OMR	OMR	Recurring	Direct	Cost	\$15 000	\$63 871
Benefits	DRB	Response & Recovery Loss Reduction	Recurring	Direct	Benefit	\$9713	\$20 813
	DRB	Total Direct Loss Reduction	Recurring	Direct	Benefit	\$2 198 382	\$2 198 382
	DRB	Indirect Loss Reduction	Recurring	Indirect	Benefit	\$88 800	\$94 350
	DRB	Fatalities Averted	Recurring	Indirect	Benefit	\$97 500	\$105 000
Externality	NDRB	Stormwater Management	Recurring	Externality	Benefit	\$200 000	
	NDRB	Value of Greenspace	Recurring	Externality	Benefit	\$9880	
	NDRB	Increase in Visitors	Recurring	Externality	Benefit	\$1 468 200	
	NDRB	Downstream Flooding	Recurring	Externality	Benefit		-\$113 886
	NDRB	Tax Revenue Loss	Recurring	Externality	Benefit	-\$367 042	-\$120 042

These costs map to E3 BCN objects as shown in Table 4-51 and Table 4-52:

Table 4-51 BCN Object Values – Buyout

VarName	Purchase of Homes	Structure Demolition	Greenspace Construction	Indirect Costs	Total OMR Costs	Stormwater Management	Total Loss Reduction	Indirect Loss Reduction	Response & Recovery Loss Reduction	Fatalities Averted	Tax Revenue	Value of Greenspace	Increase in Visitors
altID	0	0	0	0	0	0	0	0	0	0	0	0	0
bcnID	0	1	2	3	4	5	6	7	8	9	10	11	12
bcnType	Cost	Cost	Cost	Cost	Cost	Benefit	Benefit	Benefit	Benefit	Benefit	Benefit	Benefit	Benefit
bcnSubType	Direct	Direct	Direct	Indirect	Direct	Externality	Direct	Indirect	Direct	Indirect	Externality	Externality	Externality
bcnName	Purchase of Homes Buyout	Structure Demolition Buyout	Greenspace Construction Buyout	Indirect Costs Buyout	Total OMR Costs Buyout	Stormwater Management Buyout	Total Loss Reduction Buyout	Indirect Loss Reduction Buyout	Res-Rec Loss Reduction Buyout	Fatalities Averted Buyout	Tax Revenue Buyout	Value of Greenspace Buyout	Increase in Visitors Buyout
bcnTag	Initial Investment	Initial Investment	Initial Investment	Initial Investment	OMR	NDRB	DRB	DRB	DRB	DRB	NDRB	NDRB	NDRB
initialOcc	0	0	0	0	1	1	1	1	1	1	1	1	1
bcnRealBool	True	True	True	True	True	True	True	True	True	True	True	True	True
bcnInvestBool	True	True	True	True	True	False	False	false	false	False	False	False	False
bcnLife	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
rvBool	False	False	False	False	False	false	false	False	False	False	False	False	False
rvOnly	false	false	false	false	false	false	false	false	false	false	false	false	false
recurBool	False	False	False	False	True	True	True	True	True	True	True	True	True
recurInterval	Null	Null	Null	Null	1	1	1	1	1	1	1	1	1
recurVarRate	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
recurVarValue	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
recurEndDate	Null	Null	Null	Null	75	75	75	null	null	null	null	Null	Null
valuePerQ	59897500	5400000	200000	1680000	15000	200000	2198381.82	88800.216	9712.5245	97500	-367042	9880	9880
quant	1	1	1	1	1	1	1	1	1	1	1	1	1
quantVarRate	Null	Null	Null	Null	null	null	null	null	null	null	null	null	null
quantVarValue	Null	Null	Null	Null	null	null	null	null	null	null	null	null	null
quantUnit	null	null	null	null	null	null	null	null	null	null	null	null	null

Table 4-52 BCN Object Values – Levee

Variable Name	Purchase of Homes	Structure Demolition	Levee Design	Levee Construction	Indirect Costs	Total OMR Costs	Downstream Flooding	Total Loss Reduction	Indirect Loss Reduction	Response & Recovery Loss Reduction	Fatalities Averted	Tax Revenue
altID	1	1	1	1	1	1	1	1	1	1	1	1
bcnID	13	14	15	16	17	18	19	20	21	22	23	24
bcnType	Cost	Cost	Cost	Cost	Cost	Cost	Benefit	Benefit	Benefit	Benefit	Benefit	Benefit
bcnSubType	Direct	Direct	Direct	Direct	Indirect	Direct	Externality	Direct	Indirect	Direct	Indirect	Externality
bcnName	Purchase of Homes Levee	Structure Demolition Levee	Levee Design	Levee Construction	Indirect Costs Levee	Total OMR Costs Levee	Downstream Flooding Levee	Total Loss Reduction Levee	Indirect Loss Reduction Levee	Res-Rec Loss Reduction Levee	Fatalities Averted Levee	Tax Revenue Levee
bcnTag	Initial Investment	Initial Investment	Initial Investment	Initial Investment	Initial Investment	OMR	NDRC	DRB	DRB	DRB	DRB	NDRC
initialOcc	0	0	0	0	0	1	1	1	1	1	1	1
bcnRealBool	true	true	true	true	true	true	true	true	true	true	true	true
bcnInvestBool	True	True	True	True	True	true	False	False	false	false	False	False
bcnLife	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
rvBool	False	False	False	False	False	False	false	false	False	False	False	False
rvOnly	false	false	false	false	false	false	false	false	false	false	false	false
recurBool	False	False	False	False	False	True	True	True	True	True	True	True
recurInterval	Null	Null	Null	Null	Null	1	1	1	1	1	1	1
recurVarRate	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
recurVarValue	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
recurEndDate	Null	Null	Null	Null	Null	75	75	75	null	null	null	null
valuePerQ	31590000	2400000	2250000	15000000	4500000	63871	-113886	2198381.82	94350.2295	20812.5525	105000	-120042
quant	1	1	1	1	1	1	1	1	1	1	1	1
quantVarRate	Null	Null	Null	Null	Null	null	null	null	null	null	null	null
quantVarValue	Null	Null	Null	Null	Null	null	null	null	null	null	null	null
quantUnit	null	null	null		null	null	null	null	null	null	null	null

Table 4-53 shows the comparison of the E3 results to those from EDGe\$, which are identical because there are no rounding differences in the calculations as seen in the LCCA examples from BLCC. The only exception is the IRR, which have minimal differences due to the different calculation method implemented for the root-finding algorithm. For additional details on the assumptions and results of this example, please see the Economic Decision Guide User Manual [19].

Table 4-53 Summary Results Comparison – Hurricane Resilience

Results Comparison	Buyout		Levee	
	EDGe\$	E3	EDGe\$	E3
NPV				
Total Benefits	\$70 571 714	70 571 714	\$41 607 055	\$41 607 055
Total Costs	\$67 463 182	67 463 182	\$56 956 453	\$56 956 453
Net Benefits	\$3 108 532	\$3 108 532	-\$15 349 398	-\$15 349 398
Benefit-to-Cost Ratio	1.05	1.05	0.73	0.73
IRR	4.8 %	5.3 %	3.6 %	3.5%
Note: The differences in the estimates are due to EDGe\$ using continuous discounting while E3 using discrete end-of-year discounting. BCR assumes that all costs are investment.				

4.5.2.1.2 Riverbend Flooding

This example is based on the case study in Section 3.2 of the EDGe\$ User Manual [19] that considers the benefits and costs of earthquake impact mitigation strategies for Riverbend, a small city of 50 000 people located in a valley along Central River, and the neighboring city of Fallsborough. The two cities are linked by a four-lane interstate bridge that represents the only route for traffic into the city. The bridge routinely fails to meet the traffic demand during peak hours and is sensitive to earthquake events.

Therefore, two alternatives are being considered to increase community resilience against seismic event hazards: (1) retrofit the existing bridge or (2) construct a new bridge. Consideration of seismic events was driven by the known hazards in the region and the potential loss of life, infrastructure damage, and economic impacts if a disaster occurs.

This example assumes the following:

- Planning Horizon: 50 years
- Real Discount Rate: 3.00%
- Interest Rate: 3.00%
- Inflation Rate: 2.00%
- MARR: 4.00%
- Reinvestment Rate: 5.00 %
- Disaster Rate: Every 25 years
- Disaster Magnitude: 0 % of build cost
- Statistical Value of a Life: \$ 7 500 000

This information maps to the E3 Analysis object as shown in Table 4-54:

Table 4-54 Analysis Object Values

Var Name	Var Type	Format/Values	Req
analysisType	String	BCA	X
projType	String	Infrastructure	
objToReport	List of Strings	[FlowSummary, MeasureSummary]	X
studyPeriod	Int	50	X
baseDate	Date	2012, 04, 23	X
serviceDate	Date	2013, 04, 23	
timestepVal	String	Year	X
timestepComp	String	Continuous	X
outputRealBool	Boolean	True	X
interestRate	Float	0.03	
dRateReal	Float	0.03	X
dRateNom	Float	Null	
inflationRate	Float	0.02	X
Marr	Float	0.04	X
reinvestRate	Float	0.05	X
incomeRateFed	Float	0.26	
incomeRateOther	Float	0.26	
noAlt	Int	2	X
baseAlt	Int	0	X
location	List of Strings	United States, MD, 20879	

The existing bridge is scheduled and budgeted for a deck replacement in 10 years, creating an opportunity to upgrade the bridge to be more resilient to seismic events. To upgrade the bridge, it must be closed to emergency services and regular traffic. There are additional vehicle-hours from rerouting as well as impacts on emergency vehicles. Heavier traffic on alternative routes will decrease the life of those roads.

The new bridge would be built with an offset alignment from the original bridge and according to current seismic codes and a design life of 125 years. The original bridge would remain in service but would no longer be useable should a seismic event occur, pushing all traffic to be maintained by the new bridge. Sharing traffic between the bridges will reduce traffic during peak hours that would benefit long-term economic development. Apart from the immediate benefits, the new bridge would be used to carry traffic when the old bridge eventually needs to be replaced and provide pedestrian access.

The benefits and costs of each alternative are shown in Table 4-55. Costs include initial investment costs and future recurring OMR costs treated as investment costs with zero residual value (for consistency with EDGe\$) of the alternative. Disaster related benefits include response and recovery costs, direct and indirect loss reductions, and reduced fatalities. For this example, there are no externalities for the retrofit option while there are positive benefits of the new bridge including reductions in greenhouse gas and water pollution, reduced commute times, and improved community linkage.

Table 4-55 Benefits and Costs by Alternative

Desc.	Tag	Object Name	Recurr.	Sub-Type	Type	Retrofit	New Bridge
Costs	Initial Investment	Direct Cost	One-time	Direct	Cost	\$3 000 000	
	Initial Investment	Indirect Cost	One-time	Indirect	Cost	\$500 000	
	Initial Investment	Additional Roadwork Direct Cost	One-time	Direct	Cost		\$2 500 000
	Initial Investment	Bridge Construction Direct Cost	One-time	Direct	Cost		\$4 250 000
	Initial Investment	Additional Roadwork Indirect Cost	One-time	Indirect	Cost		\$120 000
	Initial Investment	Bridge Construction Indirect Cost	One-time	Indirect	Cost		\$175 000
	OMR	Additional Roadwork OMR	Recurring	Direct	Cost		\$3 710
	OMR	New Bridge OMR	Recurring	Direct	Cost		\$25 000
Benefits	DRB	Indirect Loss Reduction	One-time	Indirect	Benefit	\$80 000	\$140 000
	DRB	Direct Loss Reduction	One-time	Direct	Benefit	\$10 400	
	DRB	Response and Recovery	Recurring	Direct	Benefit	\$24 000	\$40 000
	DRB	Fatalities Averted	Recurring	Indirect	Benefit	\$30 000	\$60 000
Externalities	NDRB	Greenhouse Gas Emissions	Recurring	Externality	Benefit		\$77 329
	NDRB	Water Pollution	Recurring	Externality	Benefit		\$39 081
	NDRB	Better Linking of Communities	Recurring	Externality	Benefit		\$39 799
	NDRB	Reduced Commute Time	Recurring	Externality	Benefit		\$100 000

These costs map to E3 BCN objects as shown in Table 4-56 and Table 4-57:

Table 4-56 BCN Object Values – Retrofit

VarName	Direct Cost	Indirect Costs	Indirect Loss Reduction	Direct Loss Reduction	Response & Recovery	Fatalities Averted
altID	0	0	0	0	0	0
bcnID	0	1	2	3	4	5
bcnType	Cost	Cost	Benefit	Benefit	Benefit	Benefit
bcnSubType	Direct	Indirect	Indirect	Indirect	Direct	Indirect
bcnName	Retrofit Direct Cost	Retrofit Indirect Cost	Retrofit Indirect Loss Reduction	Retrofit Direct Loss Reduction	Retrofit Response and Recovery	Fatalities Averted Retrofit
bcnTag	Initial Investment	Initial Investment	DRB	DRB	DRB	DRB
initialOcc	0	0	1	1	1	1
bcnRealBool	true	true	true	true	true	true
bcnInvestBool	True	True	false	false	false	False
bcnLife	Null	Null	Null	Null	Null	Null
rvBool	False	False	False	False	False	False
rvOnly	false	false	false	false	false	false
recurBool	False	False	True	True	True	True
recurInterval	Null	Null	1	1	1	1
recurVarRate	Null	Null	Null	Null	Null	Null
recurVarValue	Null	Null	Null	Null	Null	Null
recurEndDate	Null	Null	50	50	null	null
valuePerQ	3000000	500000	80000	80000	24000	30000
quant	1	1	1	1	1	1
quantVarRate	Null	Null	null	null	null	null
quantVarValue	Null	Null	null	null	null	null
quantUnit	null	null	null	null	null	null

Table 4-57 BCN Object Values – New Bridge

VarName	Additional Roadwork	Bridge Construction	Additional Roadwork Indirect Cost	Bridge Construction Indirect Cost	Additional Roadwork OMR	New Bridge OMR	Greenhouse Gas Emissions	Water Pollution	Better Linking of Communities	Bridge Indirect Loss Reduction	Response & Recovery	Fatalities Averted	Reduced Commute Time
altID	1	1	1	1	1	1	1	1	1	1	1	1	1
bcnID	6	7	8	9	10	11	12	13	14	15	16	17	18
bcnType	Cost	Cost	Cost	Cost	Cost	Cost	Benefit	Benefit	Benefit	Benefit	Benefit	Benefit	Benefit
bcnSubType	Direct	Direct	Indirect	Indirect	Direct	Direct	Externality	Externality	Externality	Indirect	Direct	Indirect	Externality
bcnName	Additional Roadwork Direct Cost	Bridge Construction Direct Cost	Additional Roadwork Indirect Cost	Bridge Construction Indirect Cost	Additional Roadwork OMR	New Bridge OMR	Greenhouse Gas Emissions	Water Pollution	Better Linking of Communities	Bridge Indirect Loss Reduction	New Bridge Response and Recovery	New Bridge Fatalities Averted	Reduced Commute Time
bcnTag	Initial Investment	Initial Investment	Initial Investment	Initial Investment	OMR	OMR	NDRB	NDRB	NDRB	DRB	DRB	DRB	NDRB
initialOcc	0	0	0	0	1	1	1	1	1	1	1	1	1
bcnRealBool	true	true	true	true	true	true	true	true	true	true	true	true	true
bcnInvestBool	True	True	True	True	true	true	False	False	False	false	false	False	False
bcnLife	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
rvBool	False	False	False	False	False	False	false	false	false	False	False	False	false
rvOnly	false	false	false	false	false	false	false	false	false	false	false	false	false
recurBool	False	False	False	False	True	True	True	True	True	True	True	True	True
recurInterval	Null	Null	Null	Null	1	1	1	1	1	1	1	1	1
recurVarRate	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
recurVarValue	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
recurEndDate	Null	Null	Null	Null	50	50	50	50	50	50	null	null	50
valuePerQ	2500000	4250000	120000	175000	3710	25000	77329	39081	39799	140000	40000	60000	100000
quant	1	1	1	1	1	1	1	1	1	1	1	1	1
quantVarRate	Null	Null	Null	Null	null	null	null	null	null	null	null	null	null
quantVarValue	Null	Null	Null	Null	null	null	null	null	null	null	null	null	null
quantUnit	null	null	null	null	null	null	null	null	null	null	null	null	null

The results from E3 are identical to the EDGe\$-based calculations as shown in Table 4-58 because there are no differences from rounding as seen in the BLCC examples for LCCA. For additional details on the assumptions and results of this example, please see the Economic Decision Guide User Manual [19].

Table 4-58 Summary Results Comparison – Riverbend Flooding

Results Comparison	Retrofit		New Bridge	
NPV	EDGe\$	E3	EDGe\$	E3
Total Benefits	\$3 683 524	\$3 683 524	\$12 657 879	\$12 657 879
Total Costs	\$3 500 000	\$3 500 000	\$7 777 368	\$7 777 368
Net Benefits	\$183 523	\$183 523	\$4 880 511	\$4 880 511
Benefit-to-Cost Ratio	1.052	1.052	1.63	1.63

4.5.2.1.3 Wildland Urban Interface (WUI)

This example is based on the case study in Section B.3 of the EDGe\$ User Manual [19] that considers the benefits and costs of reducing the recurrence rate of a disaster using wildfire hazard management practices.

Assume several towns in a WUI area recently had a wildland fire burn near them with minimal impacts (minor smoke issues, no evacuation, no reported health impacts or property damage). The fire ended when a large rainstorm passed through. In examining the impacts of the fire, one major issue was that it burned a riparian forest, which led to a large amount of runoff into the local river. The town relies on this river for drinking water, and the large amount of runoff has caused substantial issues with their water treatment plant.

Worried that a larger fire could inhibit the ability of the treatment plant to provide water, a proposal was put forward that the towns should seek assistance in managing the forest in their area. This would include an organized reconstruction (reseeding and run-off barriers) of the riparian forest and management (prescribed burns) of adjacent forested areas to lessen the chances of a wildfire. Because initial reseeding must occur regardless of whether mitigation plans are implemented, only the reduction in reseeding cost in the future need to be entered into the analysis.

The baseline is to do nothing, and therefore has zero costs or benefits associated with this alternative. The alternative is to implement mitigation strategies designed to reduce the likelihood wildland urban interface (WUI) fires.

This example assumes the following:

- Planning Horizon: 50 years
- Real Discount Rate: 4.00%
- Interest Rate: 3.00%
- Inflation Rate: 2.00%
- MARR: 4.00%

- Reinvestment Rate: 5.00 %
- Disaster Rate: Every 25 years
- Disaster Magnitude: 0 % of build cost
- Statistical Value of a Life: \$750 000

This information maps to the E3 Analysis object as shown in Table 4-59:

Table 4-59 Analysis Object Values

Var Name	Var Type	Format/Values	Req
analysisType	String	BCA	X
projType	String	Infrastructure	
objToReport	List of Strings	[FlowSummary, MeasureSummary]	X
studyPeriod	Int	50	X
baseDate	Date	2012, 04, 23	X
serviceDate	Date	2013, 04, 23	
timestepVal	String	Year	X
timestepComp	String	Continuous	X
outputRealBool	Boolean	True	X
interestRate	Float	0.03	
dRateReal	Float	0.05	X
dRateNom	Float		
inflationRate	Float	0.02	X
Marr	Float	0.04	X
reinvestRate	Float	0.05	X
incomeRateFed	Float	Null	
incomeRateOther	Float	null	
noAlt	Int	2	X
baseAlt	Int	0	X
location	List of Strings	United States, MD, 20879	

The benefits and costs of WUI management plan are shown in Table 4-60. Costs include initial investment costs and OMR costs treated as investment costs with zero residual value (for consistency with EDGe\$). Benefits include sludge removal cost, water treatment cost, reseeding cost, and indirect loss reductions. Externalities can be both positive (benefits) and negative (costs) and can be one-time or recurring impacts. In this example, there are one-time external benefits for improved river health and a recurring external cost of lost recreational value from limitation on access to the forested area, which are treated as a negative benefit to align with the EDGe\$ example. Note that this value could also be inputted as a positive cost and obtain the same results.

Table 4-60 Benefits and Costs by Alternative

Desc.	Tag	Object Name	Recurr.	Sub-Type	Type	Mitigation
Costs	Initial Investment	Straw Wattles	One-time	Direct	Cost	\$6 250 860
	Initial Investment	Indirect Costs	One-time	Indirect	Cost	\$625 086
	OMR	Total OMR costs	Recurring	Direct	Cost	\$28 430
Benefits	DRB	Sludge removal cost reduction	Recurring	Direct	Benefit	\$68 000
	DRB	Water Treatment Chemical Cost Reduction	Recurring	Direct	Benefit	\$3343
	DRB	Reseeding Cost Reduction	Recurring	Direct	Benefit	\$7200
	DRB	Indirect Loss Reduction	Recurring	Indirect	Benefit	\$23 543
Externality	NDRB	Increased River Health (Watershed)	One-time	Externality	Benefit	\$3 780 000
	NDRB	Increased River Health (Salmon)	One-time	Externality	Benefit	\$4 620 000
	NDRB	Recreational Value	Recurring	Externality	Benefit	-\$51 850
Note: Base Case is zero benefits and costs						

These costs map to E3 BCN objects as shown in Table 4-61:

Table 4-61 BCN Object Values – Mitigation

VarName	Straw Wattles	Indirect Costs	Total OMR Costs	Indirect Loss Reduction	Sludge Removal Cost	Water Treatment Chemical Cost	Re seeding	Recreationa l Value	River Health (Salmon)	River Health (Watershed)
altID	1	1	1	1	1	1	1	1	1	1
bcnID	0	1	2	3	4	5	6	7	8	9
bcnType	Cost	Cost	Cost	Benefit	Benefit	Benefit	Benefit	Benefit	Benefit	Benefit
bcnSubType	Direct	Indirect	Direct	Indirect	Direct	Direct	Direct	Externality	Externality	Externality
bcnName	Straw Wattles	Indirect Costs	Total OMR Costs	Indirect Loss Reduction	Sludge Removal Cost	Water Treatment Chemical Cost	Re seeding	Recreational Value	River Health (Salmon)	River Health (Watershed)
bcnTag	Null	null	OMR	DRB	DRB	DRB	DRB	NDRB	NDRB	NDRB
initialOcc	0	0	1	1	1	1	1	1	1	1
bcnRealBool	true	true	true	true	true	true	true	true	true	true
bcnInvestBool	True	True	true	false	false	false	false	False	False	False
bcnLife	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
rvBool	False	False	False	False	False	False	False	False	False	False
rvOnly	false	false	false	false	false	false	false	false	false	false
recurBool	False	False	True	True	True	True	True	True	False	False
recurInterval	Null	Null	1	1	1	1	1	1	1	1
recurVarRate	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
recurVarValue	Null	Null	Null	Null	Null	Null	Null	Null	Null	Null
recurEndDate	Null	Null	50	null	null	null	null	Null	Null	Null
valuePerQ	6250860	625086	28430	23543.2	68000	3343.04	7200	51850	4620000	3780000
quant	1	1	1	1	1	1	1	1	1	1
quantVarRate	Null	Null	null	null	null	null	null	null	null	null
quantVarValue	Null	Null	null	null	null	null	null	null	null	null
quantUnit	null	null	null	null	null	null	null	null	null	null

The results from E3 are identical to the EDGe\$ calculations as shown in Table 4-62. For additional details on the assumptions and results of this example, please see the Economic Decision Guide User Manual [19]. Note that if the user desires calculation of supplemental measure results with and without externalities, it is necessary to submit two json strings to E3.

Table 4-62 Summary Results Comparison - WUI

Results Comparison		Mitigation	
Measure		EDGe\$	E3
Total NPV	Total Benefits	\$7 380 709	\$7 380 709
	Total Costs	\$7 015 776	\$7 015 776
	Net Benefits	\$364 934	\$364 934
	Benefit-to-Cost Ratio	1.05	1.05
	IRR	4.79 %	4.6 %
Assumes all costs are investment.			

4.5.2.2 Continuous Discounting with Sensitivity

This uses the same example from Section 4.5.2.1.3 (WUI) but adds sensitivity objects to see the variability of the discount rate (a global variable) as well as the value per quantity for two BCN objects (non-global variables) to validate a percent and gross change to the value. For the discount rate, as with any sensitivity object for a global variable, has null values for altID, bcnID, and bcnObj because the change is occurring in an Analysis Object parameter and not in a BCN object.

Table 4-63 and Table 4-64 show the results from the two sensitivity analyses. First, increase the discount rate from 3 % to 6 % (increase quantity value by “gross” “0.03”). Note that “varName” is “discountRate,” which changes either “dRateReal” or “dRateNom” depending on whether it’s a real or nominal analysis, respectively. The increase in the real discount rate from 3 % to 6 % leads to future benefits and costs decreasing in net present value terms and lowering the net benefits as shown in Table 4-63.

Table 4-63 Summary Results Comparison – Riverbend Flooding with Sensitivity – Discount Rate (Global Variable)

Results Comparison	Retrofit		New Bridge	
	3 %	6 %	3 %	6 %
NPV				
Total Benefits	\$3 683 524	\$2 218 926	\$12 657 879	\$7 625 009
Total Costs	\$3 500 000	\$3 500 000	\$7 777 368	\$7 486 173
Net Benefits	\$183 524	-\$1 281 074	\$4 880 511	\$138 835
Benefit-to-Cost Ratio	1.052	0.63	1.63	1.02

Second, double the cost of the bridge (increase quantity value by “percent” of “100.0”). Doubling the initial direct costs of constructing the new bridge leads to a direct decrease

in net benefits equivalent to the incremental increase in construction costs as shown in Table 4-64.

Table 4-64 Summary Results Comparison – Riverbend Flooding with Sensitivity – New Bridge Construction Costs

Results Comparison	Retrofit	New Bridge		
NPV	Original	Original	0.5x New Bridge Cost	2x New Bridge Cost
Total Benefits	\$3 683 524	\$12 657 879	\$12 657 879	\$12 657 879
Total Costs	\$3 500 000	\$7 777 368	\$ 5 652 368	\$ 12 027 368
Net Benefits	\$183 523	\$4 880 511	\$7 005 510	\$630 510
Benefit-to-Cost Ratio	1.052	1.63	2.24	1.05

4.5.2.3 Continuous Discounting with Uncertainty

Uncertainty is not available in the current version of E3. The next release of E3, which is being redeveloped in Java, will include uncertainty capabilities. See Section 3.3 for additional details.

4.5.3 Sustainable Manufacturing

NIST Advanced Manufacturing Series 200-11 [21] “presents techniques for evaluating manufacturing investments from the perspective of environmentally sustainable manufacturing by pairing economic methods of investment analysis with environmental aspect of manufacturing, including manufacturing processes.” Example 10 in [21] provides analysis of multiple investment alternatives across a range of metrics: MEER, Net Present Value (NPV), Internal Rate of Return (IRR), and pairwise comparisons. Below are the details of Example 10 that are used for validating E3.

Analysis assumptions:

- 10-year study period
- 4 % real discount rate
- \$2000 environmental hurdle rate
- \$2000 maximum environmental expenditure rate (MEER)

Costs by alternative are as follows:

- Base case (Alternative 0)
 - \$25 000 in machinery repairs
- Alternative 1
 - \$100 000 to purchase a new piece of machinery
 - Decrease in annual maintenance costs by \$8000
 - \$10 000 salvage value at end of study period
 - Environmental impact would be decreased by 5 %

- Alternative 2
 - \$150 000 to purchase a new piece of machinery
 - Decrease in annual maintenance costs by \$20 000
 - \$20 000 salvage value
 - Environmental impact decreased by 10 %
- Alternative 3
 - \$200 000 to purchase a high-efficiency piece of machinery
 - Decrease in annual maintenance costs by \$20 000
 - \$30 000 salvage value
 - Decrease in annual energy costs by \$4000
 - Environmental impact decreased by 20 %

Table 4-65 shows the costs and environmental impacts by alternative.

Table 4-65 Costs and Environmental Impacts

Alternative	Initial Costs		Future Savings and Reductions			
	Machinery	Repairs	Maintenance Savings	Salvage Value	Energy Savings	Environmental Impact Reduction
0	\$0	\$25 000	\$0	\$0	\$0	0 %
1	\$100 000	\$0	\$8000	\$10 000	\$0	5 %
2	\$150 000	\$0	\$20 000	\$20 000	\$0	10 %
3	\$200 000	\$0	\$20 000	\$30 000	\$4000	20 %

These costs, savings, and impact reductions map to E3 BCN objects as shown in Table 4-66 and Table 4-67.

Table 4-66 BCN Object Values – Alternative 0 and Alternative 1

VarName	Repair 0	Machinery Purchase 1	Maintenance 1	Salvage Value 1	Environmental Impact 0	Environmental Impact 1
altID	0	1	1	1	0	1
bcnID	0	1	2	3	4	5
bcnType	Cost	Cost	Cost	Cost	Nonmonetary	Nonmonetary
bcnSubType	Direct	Direct	Direct	Direct	Externality	Externality
bcnName	Repair 0	Machinery Purchase 1	Maintenance 1	Salvage Value 1	Environmental Impact 0	Environmental Impact 1
bcnTag	OMR	Initial Investment	OMR	Salvage Value	Environmental Impact	Environmental Impact
initialOcc	0	0	1	10	0	1
bcnRealBool	true	true	true	true	true	true
bcnInvestBool	false	true	false	True	False	False
bcnLife	Null	null	null	Null	Null	Null
rvBool	False	false	false	true	False	False
rvOnly	false	false	false	true	false	false
recurBool	False	false	true	False	True	True
recurInterval	Null	Null	1	Null	1	1
recurVarRate	Null	Null	Null	Null	Null	Null
recurVarValue	Null	Null	Null	Null	Null	Null
recurEndDate	Null	Null	Null	Null	Null	Null
valuePerQ	25000	100000	-8000	-10000	1	1
quant	1	1	1	1	100	95
quantVarRate	Null	Null	Null	Null	Null	Null
quantVarValue	Null	Null	Null	Null	Null	Null
quantUnit	Null	Null	Null	Null	Units	Units

Table 4-67 BCN Object Values – Alternative 2 and Alternative 3

VarName	Machinery Purchase 2	Machinery Purchase 3	Maintenance 2/3	Salvage Value 2	Salvage Value 3	Energy 3	Environmental Impact 2	Environmental Impact 3
altID	2	3	[2,3]	2	3	3	2	3
bcnID	6	7	8	9	10	11	12	13
bcnType	Cost	Cost	Cost	Cost	Cost	Cost	Nonmonetary	Nonmonetary
bcnSubType	Direct	Direct	Direct	Direct	Direct	Direct	Externality	Externality
bcnName	Machinery Purchase 2	Machinery Purchase 3	Maintenance 2/3	Salvage Value 2	Salvage Value 3	Energy 3	Environmental Impact 2	Environmental Impact 3
bcnTag	Initial Investment	Initial Investment	OMR	Salvage Value	Salvage Value	Energy	Environmental Impact	Environmental Impact
initialOcc	0	0	1	10	10	1	1	1
bcnRealBool	true	true	true	true	true	true	true	true
bcnInvestBool	true	true	false	True	True	False	False	False
bcnLife	null	null	null	Null	Null	Null	Null	Null
rvBool	false	false	false	True	True	False	False	False
rvOnly	false	false	false	true	true	false	false	false
recurBool	false	false	true	False	False	True	True	True
recurInterval	Null	Null	1	Null	Null	1	Null	Null
recurVarRate	Null	Null	Null	Null	Null	Null	Null	Null
recurVarValue	Null	Null	Null	Null	Null	Null	Null	Null
recurEndDate	Null	Null	Null	Null	Null	null	Null	Null
valuePerQ	150000	200000	-20000	-20000	-30000	-4000	1	1
quant	1	1	1	1	1	1	90	80
quantVarRate	Null	Null	Null	Null	Null	Null	Null	Null
quantVarValue	Null	Null	Null	Null	Null	Null	Null	Null
quantUnit	Null	Null	Null	Null	Null	Null	Units	Units

The environmental impacts (non-monetary BCN objects) could be submitted in numerous ways to get the desired comparison. In this example, the baseline is assumed to have 100

“units” of the environmental impact annually while the alternatives have a fraction of those 100 units annually. The units are aggregated over 10 years with the baseline having a total of 1000 units. Alternative 1, 2, and 3 have a total of 950, 900, and 800 units, respectively.

The results from the E3 API were compared to the results in the example in Table 4-68 to confirm consistency between E3 and BLCC. Ref. [21] does not provide annual cash flows or total NPV in the example, and instead provide the net savings, IRR, and percentage change in environmental impacts. The results are identical for the NPV (which is reverse engineered for Ref. [21]), net savings, IRR, and environmental impact percentage changes for all alternatives.

Table 4-68 Summary Results Comparison

Alt.	NPV (\$)		NS		IRR		Environmental Impact	
	Thomas (XXXX)	E3	Thomas (XXXX)	E3	Thomas (XXXX)	E3	Thomas (XXXX)	E3
0	\$25 000	\$25 000	-	-	-	-	-	-
1	\$28 357	\$28 357	-\$3357	-\$3357	3.2 %	NaN	-5.0 %	-5.0 %
2	-\$25 729	-\$25 729	\$50 729	\$50 729	11.0 %	3.0 %	-10.0 %	-10.0 %
3	-\$14 928	-\$14 928	\$39 928	\$39 928	8.0 %	8.0 %	-20.0 %	-20.0 %

Alternative 2 and Alternative 3 are preferred when compared to the base case as well as Alternative 1 because the total NPV costs and the environmental impacts are lower. Alternative 1 has a tradeoff where total NPV costs are higher than the base case, but the environmental impacts are lower. Choosing between Alternative 2 and Alternative 3 requires a pairwise comparison, as there is a tradeoff between the costs and environmental impacts.

When combined with the provided environmental hurdle rate and MEER, the E3 results provide all the necessary information to complete the additional calculations to make the pairwise comparison. In this example, the net present value per percent change in environmental impact (NPVP) needs to be calculated, which is the dollar trade-off per percent reduction in environmental impact.

Relative to Alternative 2, Alternative 3 has higher costs (additional \$10 801) and 10 % lower environmental impacts. The average cost per percent reduction in environmental impacts (or Net Present Value per Percent Increase) is \$1080, which is lower than the MEER of \$2000. Therefore, Alternative 3 is preferred to Alternative 2.

Note that E3 generates three measures that could be used for trade-offs between monetary and non-monetary results:

- nsPercQuant is the Net Savings Per Percent Change in Quantity
- nsDeltaQuant is the Net Savings Per Unit Change in Quantity
- nsElasticityQuant is the Net Savings Elasticity Per unit of Quantity Change

In this example, the Net Savings Per Percent Change in Quantity could be used in this example if the example only compared Alternative 3 to Alternative 2 as the base case. This value would be equivalent to the NPVP. Net Savings of -\$10 801 and Percent Change in Quantity of -10 % means nsPercQuant would be \$1080.

Other measures discussed in the sustainable manufacturing standard section include the Net present value elasticity (NPVE), which is equivalent to the nsElasticityQuant value.

The nsDeltaQuant could be used to compare the additional costs of a unit of emissions reduction to the social cost of those emissions. For example, an alternative that reduces greenhouse gas (GHG) emissions by 100 metrics tons at an additional cost of \$3000 could be compared to the externality cost of those GHG emissions (e.g., social cost of carbon) to determine if the additional benefits to society are worth the additional costs.

4.5.4 [PV]²

The [PV]² web application allows a user to complete a life cycle cost analysis and grid-based electricity reduction estimate of a residential solar photovoltaic system. This tool has allowed for validation of the life-cycle cost analysis calculations and all related economic measures/metrics in E3 (i.e., NPV cash flows, NS, IRR, SIR, and AIRR). Additionally, the tool allowed for validation of changes in quantity flows and non-monetary externalities (impacts that are not realized by the entity making the investment decision), specifically life cycle impact assessment of the solar photovoltaic system and its electricity production including global warming potential and other environmental impact categories. The tool currently post-processes the monetization of the global warming potential impact category using the social cost of carbon (SCC). For additional details on [PV]², please visit <https://pv2.el.nist.gov/> to access the web application and the user guide [22]. The code for the tool is also available on GitHub at <https://github.com/usnistgov/pv2>.

Additional capabilities are being considered for inclusion within [PV]² to test other aspects of the E3 functionality, specifically:

- Sensitivity analysis of key user inputs that have the greatest potential impact on the results (e.g., discount rate, study period)
- Uncertainty analysis of the user inputs considered to have the greatest uncertainty (e.g., energy price escalation rates)

4.5.5 Other Examples

Additional examples may be added in future versions of this document. Under consideration are as follows: Agency Vehicle Fleet, Product Service Life Improvements, Environmental Justice. User feedback and new use cases will determine what examples are added in the future.

5 Planned Release and Current Capabilities and Future Capabilities

5.1 Release of E3

E3 is currently available internally to NIST researchers on GitHub (<https://github.com/usnistgov/e3>) and at the E3 web API path (<https://e3.nist.gov/api/v1/>). E3 was released to the public in April 2022 through GitHub. The external facing E3 web API is currently available on a case-by-case basis for external developers by contacting the E3 development team.

The next release of E3 will supersede the current version and include several fundamental changes as well as an expansion of capabilities. First, E3 is being rewritten in Java to take advantage of existing libraries that simplify and improve performance of E3 calculations. This improved performance is important for the new uncertainty capabilities that will be available in the new version.

The current python-based version of E3 will not be further developed nor will a web API be maintained for internal or external use. However, the code will remain available on GitHub. At the time of release of the new version, the legacy code will be moved within the GitHub directory. The next release of E3 (Java-based) will be the focus of future development and support, and the E3 web API path based on the new version will be <https://e3.nist.gov/api/v2/>.

5.2 Current Capabilities

The current version of E3 has been designed to complete the following economic analysis and guidance is provided to map different terminology to E3:

- Life Cycle Cost Analysis (LCCA)
 - costs, savings, net savings
- Benefit-Cost Analysis (BCA) / Cost-Benefit Analysis (CBA)
 - benefits, costs, net benefits
- Profit Maximization
 - revenue, cost, profit, loss
- Cost Minimization
 - Costs, savings, net savings
- Monetary / Non-monetary Trade-Off Analysis
 - Costs, savings, net savings, cost per unit change

The above economic analysis types can use the following capabilities/features:

- Deterministic estimates
- Sensitivity testing
- Different discounting conventions
 - End-of-Year (EOY)
 - Mid-Year

- Continuous
- Nominal and real discount rates

5.3 Future Capabilities⁷

The following capabilities/features are not currently included, but will be available in the next release of E3:

- Uncertainty

The following economic analysis types are not currently included, but may be added in the future including guidance and examples on mapping to E3:

- Non-monetary to non-monetary trade-offs
- Others identified by stakeholders and users

Additional features not currently included, but may be added in the future include:

- Default datasets
 - Energy price escalation rates
 - Discount and inflation rates
- New modules
- User feedback

A few capabilities that would be useful to include in E3 in the future include the following:

- Allow alternatives with no BCN objects (e.g., baseline case)
- Allow BCNs for an alternative to include both nominal and real dollar values (e.g., loan payments in nominal dollars and other costs in real dollars)

Access to E3 is limited to those that receive a valid authentication token from NIST, which is currently provided on a limited basis for security (cybersecurity) and cost (cloud server services) reasons. Access may be expanded and improve in the future based on user demand and NIST security requirements.

E3 could be included in future versions of existing ASTM standards as a resource/supplement with examples explicit to each standard.

⁷ These lists are subject to change as development progresses

References

- [1] ASTM (2021) *Committee E06 on Performance of Buildings - Committee Scope* (<https://www.astm.org/COMMIT/SCOPES/E06.htm>). Available at
- [2] ASTM (2021) *Standard Terminology of Building Economics*, ASTM Designation E833 – 14(2021). (West Conshohocken, PA).
- [3] ASTM (2015) *Standard Guide for Selecting Economic Methods for Evaluating Investments in Buildings and Building Systems*, ASTM Designation E1185-15. (West Conshohocken, PA).
- [4] ASTM (2020) *Standard Guide for Summarizing the Economic Impacts of Building-Related Projects*, ASTM Designation E2204-15(2020)e1. (West Conshohocken, PA).
- [5] ASTM (2017) *Standard Practice for Measuring Life-Cycle Costs of Buildings and Building Systems*, ASTM Designation E917-17e1. (West Conshohocken, PA).
- [6] ASTM (2020) *Standard Practice for Measuring Net Benefits and Net Savings for Investments in Buildings and Building Systems*, ASTM Designation E1074 – 15(2020)e1. (West Conshohocken, PA).
- [7] ASTM (2020) *Standard Practice for Measuring Benefit-to-Cost and Savings-to-Investment Ratios for Buildings and Building Systems*, ASTM Designation E964 – 15(2020)e1. (West Conshohocken, PA).
- [8] ASTM (2020) *Standard Practice for Measuring Internal Rate of Return and Adjusted Internal Rate of Return for Investments in Buildings and Building Systems*, ASTM Designation E1057 – 15(2020)e1. (West Conshohocken, PA).
- [9] ASTM (2020) *Standard Practice for Measuring Payback for Investments in Buildings and Building Systems*, ASTM Designation E1121 – 15(2020)e1. (West Conshohocken, PA).
- [10] ASTM (2020) *Standard Guide for Selecting Techniques for Treating Uncertainty and Risk in the Economic Evaluation of Buildings and Building Systems*, ASTM Designation E1369-15(2020)e1. (West Conshohocken, PA).
- [11] ASTM (2018) *Standard Practice for Measuring Cost Risk of Buildings and Building Systems and Other Constructed Projects*, ASTM Designation E1946-18. (West Conshohocken, PA).
- [12] ASTM (2020) *Standard Guide for Developing a Cost-Effective Risk Mitigation Plan for New and Existing Constructed Facilities*, ASTM Designation E2506-15(2020)e1. (West Conshohocken, PA).
- [13] ASTM (2016) *Standard Practice for Applying Analytical Hierarchy Process (AHP) to Multiattribute Decision Analysis of Investments Related to Projects, Products, and Processes*, ASTM Designation E1765-16e1. (West Conshohocken, PA).
- [14] CFR (2018) *10 Code of Federal Regulations (CFR) 436, Subpart A, Methodology and Procedures for Life-Cycle Cost Analysis*. (<https://www.ecfr.gov/cgi-bin/text-idx?SID=1772217352e2b3956b37b48739ebd676&mc=true&node=pt10.3.436&rgn=div5>).
- [15] Kneifel J , Webb D (2020) *LIFE CYCLE COSTING MANUAL for the Federal Energy Management Program* (<https://nvlpubs.nist.gov/nistpubs/hb/2020/NIST.HB.135-2020.pdf>), (Technology NloSa).
- [16] ASTM (2018)– *Standard Guide for Developing Cost-Effective Community Resilience Strategies*, ASTM Designation E3130-18 (ASTM International, <https://www.astm.org/Standards/E3130.htm>).

-
- [17] ASTM (2020) Standard Practice for Performing Value Engineering (VE)/Value Analysis (VA) of Projects, Products and Processes, ASTM Designation E1699-14(2020). (West Conshohocken, PA).
- [18] ASTM (2010) Standard Guide for Investment Analysis in Environmentally Sustainable Manufacturing, ASTM Designation E3200-21. (West Conshohocken, PA).
- [19] Helgeson JF, Webb DH, Grubb SA (2017) The Economic Decision Guide Software (EDGe\$) Tool User Guidance. (National Institute of Standards and Technology).
<https://doi.org/https://doi.org/10.6028/NIST.SP.1214>
- [20] NIST (2021) *Building Life-Cycle Cost (BLCC) 5.3 software* (National Institute of Standards and Technology (NIST) for Federal Energy Management Program (FEMP)). Available at <https://energy.gov/eere/femp/building-life-cycle-cost-programs>.
- [21] Thomas D, Kandaswamy A, Butry D (2021) Guide for Environmentally Sustainable Investment Analysis Based on ASTM E3200. (National Institute of Standards and Technology, Gaithersburg, MD).
- [22] Kneifel J, Webb D, Donmoyer L (2022) Present Value of Photovoltaics - [PV]2 - User Guide. (National Institute of Standards and Technology (NIST), Gaithersburg, MD).
<https://doi.org/https://doi.org/10.6028/NIST.TN.2219>