Abstract

A Reference Architecture captures the essence of the architecture of a collection of systems. The purpose of a Reference Architecture is to provide guidance for the development of architectures for new versions of the system or extended systems and product families.

We provide guidelines for the content of a Reference Architecture and the process to create and maintain it. A Reference Architecture is created by capturing the essentials of existing architectures and by taking into account future needs and opportunities, ranging from specific technologies, to patterns to business models and market segments.
1. general introduction
2. level of abstraction
3. content
4. summary
Why Reference Architectures?

When to Use Reference Architectures?

What do Reference Architectures contain?

How to use Reference Architectures?

What are inputs of a Reference Architecture?

Criteria for a good Reference Architecture.
Graph of objectives of Reference Architectures

- **Increased complexity, scope, size**
  - Facilitate multi-site, multi-organization, multi-vendor systems
  - System creation and life-cycle support

- **Increased dynamics, integration**
  - Effectively create new: products, product lines, product portfolio

- **Achieve interoperability between many different and evolving systems**

**Managing synergy**
- Effectively create new: products, product lines, product portfolio

**Providing guidance, e.g. architecture principles, best practices**
- Providing an architecture baseline and an architecture blueprint

**Capturing and sharing (architectural) patterns**
- Providing a common lexicon and taxonomy
- Providing a common (architectural) vision
- Providing modularization and the complementary context

**Articulation of domain and realization concepts**
- Explicit modeling of functions and qualities above systems level
- Explicit decisions about compatibility, upgrade and interchangeability.
When to Use Reference Architectures

Reference Architectures facilitate the step towards product family architecting and evolvability; this often coincides with multi-* problems.

- evolvable product family architecting
- product family architecting
- mono-system architecting
- mono-system design
- mono-disciplinary engineering

Reference Architecture
multi-site multi-supplier multi-vendor

A Reference Architecture Primer
Gerrit Muller
version: 0.6
September 3, 2020
RAPwhen
RA Elaborates Mission, Vision and Strategy

A Reference Architecture Primer

Gerrit Muller

version: 0.6
September 3, 2020
SAFRAstrategy
RA = Business Arch. + Technical Arch. + Customer Context

- Customer context
  - Customer enterprise
  - Users
  - Requirements
  - Black box view

- Technical architecture
  - Design patterns
  - Technology

- Relations
  - Guidance
  - Life cycle
  - Business model

- Business architecture
Instantiation of a RA in few Transformations

A Reference Architecture Primer
Gerrit Muller
version: 0.6
September 3, 2020
SAFRAinstantiation
Inputs of a Reference Architecture

- Existing architectures
- Customer needs
- Business needs
- Product portfolio
- Future requirements

- Essence
- Architecture patterns
- Proven concepts & known problems
- Exploration & analysis
- Vision

Reference Architecture

Guides evolution

Triggers new changes
Criteria for a good Reference Architecture

- Understandable for broad set of stakeholders
- Accessible and actually read/seen by majority of the organization
- Addresses the key issues of the specific domain
- Satisfactory quality
- Acceptable
- Up-to-date and maintainable
- Adds value to the business

Customers, product managers, project managers, and engineers...
Challenge: Appropriate Level of Abstraction

Single System
Product Family in Context
Capturing the Essence
Size Considerations:
  What is the appropriate level of abstraction?
  How many details?
Decomposition of Large Documents

or

Reference Architecture

A Reference Architecture Primer
Gerrit Muller
version: 0.6
September 3, 2020
RAPlevelOfAbstraction
Level of Abstraction Single System

- Static system definition
- Multidisciplinary design
- System requirements

number of details

10^0
10^1
10^2
10^3
10^4
10^5
10^6
10^7

monodisciplinary

system

multidisciplinary

design

static system definition

monodisciplinary
Product Family in Context

- enterprise context
- enterprise
- stakeholders
- systems
- multidisciplinary design
- parts, connections, lines of code

$10^9$ $10^6$ $10^3$ $10^0$ $10^3$ $10^6$ $10^9$

details

number of
RA: Capturing the Essence

Reference Architecture Primer

enterprise context
enterprise
stakeholders
systems
multidisciplinary design
parts, connections, lines of code

some context details are essential
some technical details are essential
RA: level of abstraction, number of details

10^3  level of abstraction  10^6  number of details

compact reference architecture few diagrams only

extensive reference architecture many documents

market process flow key performance indicators information model decomposition concurrency & synchronization

high end market value chain industry roadmap key drivers strategic partners business models

process flow interoperability function flow key performance indicators standards

decomposition behavior functional models security supplier policy

concurrency & synchronization AAPI's shared assets

Cost of Ownership process descriptions map of systems

information model cache design

A Reference Architecture Primer
15  Gerrit Muller

version: 0.6
September 3, 2020
RAPsizeSpectrum
Size Considerations

10^3
level of abstraction

10^6
number of details

compacted reference architecture
few diagrams only

low effort to
create
maintain
read
easy to share

limited
guidance
anchor value

extensive reference architecture
many documents

significant effort to
create
maintain
read
difficult to share

great
guidance
anchor value

market
process flow
key drivers
key performance indicators
information model
concurrency & synchronization

high end market
value
industry roadmap
key drivers
strategic partners
key performance indicators
process descriptions
business models
map of systems

resource management
interoperability
function flow
standards

functional models
security
exception handling

supplier policy
API's
shared assets

concurrency & synchronization

A Reference Architecture Primer
16  Gerrit Muller
Decomposition of Large Documents

![Diagram showing the structure of large documents with recursion.]
What should be in Reference Architectures?

- Guidance from Best Practices
- Visualizations
- Structure

What content should be in Reference Architectures?
1.1 One of several prerequisites for architecture creative synthesis is the definition of **5-7 specific key drivers** that are critical for success, along with the rationale behind the selection of these items.

2.1. The essence of a system can be captured in about **10 models/views**.

2.2. A **diversity** of architecture descriptions and models is needed: languages, schemata and the degree of formalism.

2.3. The level of **formality** increases as we move closer to the implementation level.

from http://www.architectingforum.org/bestpractices.shtml
Possible useful visualizations

actual figures and references to their use at http://www.gaudisite.nl/figures/<name>.html
Ideal Structure does not exist
Synthesis, Integration, Relation oriented

1. Functional Decomposition

2. Construction Decomposition

3. Allocation

4. Infrastructure

5. Choice of integrating concepts
Checklist for RA content

customer context

- business
- financials
- stakeholders
- benefits, concerns
- concept of operations

technical architecture

- key performance parameters
- product features, functions

- core technologies
- critical resources
- design issues
- dominant patterns

relations guidance

- business model
- life cycle
- stakeholders
- benefits, concerns

business architecture
Summary of the role of Reference Architectures

- **mission**
- **vision**
- **strategy**
- **technology**
- **customers**
- **market**
- **needs**
- **opportunities**
- **elaboration**
- **guidance**
- **knowledge**
- **existing architectures**
- **multiple organizations**
- **new or evolved architectures**