The Rise of the Development Environment Architect

Peter Eeles
Chief Architect for IT
IBM Rational Worldwide Solution Delivery

peter.eeles@uk.ibm.com
Agenda

- Start
- Development environment slides
  - Architect slides
  - Development environment architect slides
- Finish
Putting a Development Environment in Context

Center of Excellence

Creates & Maintains

Development Environment

Supports

Development Project

Creates & Maintains

Software-Intensive System
A Simple Definition

<table>
<thead>
<tr>
<th>Solution Context (functionality, qualities, constraints)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
</tr>
<tr>
<td>Roles, work products, tasks, processes</td>
</tr>
<tr>
<td>Standards, guidelines, checklists etc.</td>
</tr>
<tr>
<td>Method deployment topology</td>
</tr>
<tr>
<td><strong>Tools</strong></td>
</tr>
<tr>
<td>Development tools and integrations</td>
</tr>
<tr>
<td>Development tool configurations and install scripts</td>
</tr>
<tr>
<td>Development tool deployment topology</td>
</tr>
<tr>
<td><strong>Enablement</strong></td>
</tr>
<tr>
<td>Training curriculum and courses</td>
</tr>
<tr>
<td>Mentoring materials</td>
</tr>
<tr>
<td>Enablement deployment topology</td>
</tr>
<tr>
<td><strong>Organization</strong></td>
</tr>
<tr>
<td>Organizational roles and units</td>
</tr>
<tr>
<td>Organization deployment topology</td>
</tr>
<tr>
<td><strong>Infrastructure</strong></td>
</tr>
<tr>
<td>Locations, nodes and connectivity</td>
</tr>
<tr>
<td>Supporting software (such as operating systems)</td>
</tr>
<tr>
<td><strong>Adoption</strong></td>
</tr>
<tr>
<td>Adoption plan</td>
</tr>
<tr>
<td>Techniques for driving organizational change</td>
</tr>
<tr>
<td>Environment metrics</td>
</tr>
</tbody>
</table>
Required Functionality is Realized by All Elements
Key Considerations

- **Solution Context**
  - The requirements on the development environment

- **Solution Definition**
  - The solution elements that comprise the development environment

- **Solution Deployment**
  - Concerns when deploying the environment (e.g. on projects)

- **Solution Management**
  - Concerns when supporting the operational environment

*Each “wave of change / iteration”, by definition, changes the context for the next wave*
# Solution-focused Elements

<table>
<thead>
<tr>
<th>Method</th>
<th>Solution Definition</th>
<th>Solution Deployment</th>
<th>Solution Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roles, work products, tasks, processes Standards, guidelines, checklists etc. Method deployment topology</td>
<td>Define local configuration Deploy method</td>
<td>Gather feedback on method</td>
<td></td>
</tr>
<tr>
<td>Development tools and integrations Development tool config., install scripts Development tool deployment topology</td>
<td>Perform local configuration Install tools Migrate local data</td>
<td>Backup / archive / restore data Gather feedback on tools</td>
<td></td>
</tr>
<tr>
<td>Training curriculum and courses Mentoring materials Enablement materials deployment topology</td>
<td>Perform local configuration Deploy enablement materials Train practitioners</td>
<td>Mentor practitioners Gather feedback on enablement</td>
<td></td>
</tr>
<tr>
<td>Organizational roles and units Organization deployment topology</td>
<td>Define local configuration Reorganize</td>
<td>Gather feedback on organization</td>
<td></td>
</tr>
<tr>
<td>Locations, nodes and connectivity Supporting software (such as OS)</td>
<td>Define local infrastructure Provision locations, nodes, connectivity Provision supporting software</td>
<td>Onboard / retire infra. as required Gather feedback on infrastructure</td>
<td></td>
</tr>
<tr>
<td>Adoption plan Techniques for driving org. change Environment metrics</td>
<td>Define local adoption plan Validate the environment</td>
<td>Measure environment effectiveness Gather feedback on adoption</td>
<td></td>
</tr>
</tbody>
</table>
Solution Elements are Inter-related

- Mary used the defined techniques (method) for making tools (tools) available (adoption) to defined practitioners (organization) by configuring the license server (infrastructure) appropriately.
Development Environment Definition
Ensuring a comprehensive consideration of all elements of a development environment
Agenda

- Start
- Development environment slides
- Architect slides
- Development environment architect slides
- Finish
Key Reference

The Process of Software Architecting

Peter Eeles
Peter Cripps

Foreword by Grady Booch
Architecture 101

- performs

Architect

- creates

Architecting

- results in

Architecture
Software Architecture Defined

- Architecture is the fundamental **organization** of a **system** embodied in its **components**, their **relationships** to each other, and to the **environment**, and the **principles** guiding its design and evolution. [IEEE 1471]

- The software architecture of a program or computing system is the **structure** or structures of the system, which comprise software **elements**, the externally visible properties of those elements, and the **relationships** among them. [Bass 2003]

- The software architecture of a system or a collection of systems consists of all the **important design decisions** about the software **structures** and the **interactions** between those structures that comprise the systems. The design decisions support a desired set of **qualities** that the system should support to be successful. The design decisions provide a conceptual basis for system development, support, and maintenance. [McGovern 2004]
Characteristics of a Software Architecture

- An architecture defines structure
- An architecture defines behaviour

Structure

Behavior

1: create order
1.1: get customer details
1.2: create order
1.3: place order
1: add order item
Characteristics of a Software Architecture

- An architecture balances stakeholder needs
- An architecture is influenced by its environment
- An architecture is concerned with significant elements
- An architecture may conform to an architectural style
- An architecture influences development team structure
- An architecture is present in every system
- An architecture embodies decisions based on rationale
Characteristics of a Software Architecture

- An architecture has a particular scope

'A system is] a set of resources that provide services that are used by an enterprise to carry out a business purpose or mission. System components typically consist of hardware, software, data, and workers. [Cantor 2003]
Characteristics of an Architect

- The architect is a technical leader
- The architect understands the development process
- The architect has knowledge of the business domain
- The architect has technology knowledge
- The architect has design skills
- The architect has programming skills
- The architect is a good communicator
- The architect makes decisions
- The architect is aware of organizational politics
- The architect is a negotiator
- The architect role may be fulfilled by a team

"The life of a software architect is a long and rapid succession of suboptimal design decisions taken partly in the dark."

- Philippe Kruchten
Characteristics of Architecting

- Architecting is a science
  - We can apply scientific rigor to what we do in terms of, for example, reusable assets, method, etc.

- Architecting is an art
  - There is still a need for creativity
Characteristics of Architecting

- Architecting changes emphasis over time

*Drawn on a napkin by Bran Selic for Philippe Kruchten*
Characteristics of Architecting

- Architecting spans many disciplines

All architecture is design but not all design is architecture. Architecture represents the significant design decisions that shape a system, where significant is measured by cost of change.

- Grady Booch

[A discipline is a] primary categorization mechanism for organizing tasks that define a major 'area of concern' and/or cooperation of work effort. [OpenUP 2008]
Characteristics of Architecting

- Architecting involves many stakeholders
- Architecting is involved in tradeoffs
- Architecting considers reusable assets
- Architecting is both top-down and bottom-up
Benefits of Architecting

- Architecting addresses system qualities

- Stakeholder input
  - Functional Requirements
    - Scalability Performance
    - Runtime Qualities
    - Business Constraints
  - Schedule Resources
    - Technical Constraints
    - Distribution Platforms
    - Maintainability Portability
Benefits of Architecting

- Architecting supports the planning process
Benefits of Architecting

- Architecting helps manage complexity
- Architecting ensures architectural integrity
- Architecting reduces maintenance costs
- Architecting drives consensus
- Architecting provides a basis for reuse
- Architecting supports impact analysis
Agenda

- Start
- Development environment slides
- Architect slides

Development environment architect slides
- Finish
An Architecture is concerned with Significant Elements

- Significant elements
  - Relate to some critical functionality of the system
  - Relate to some critical property of the system
  - Relate to a particular architectural challenge
  - Are associated with a particular technical risk
  - Relate to a capability that is considered to be unstable
  - Relate to some key element of the solution

- Development environment-specifics
  - Method: roles & responsibilities, work products, governance policies
  - Tools: selection, integrations, licensing
  - Enablement: curriculum
  - Organization: roles & responsibilities
  - Infrastructure: distribution, development environment packaging
An Architecture has a Particular Scope

Enterprise System

Software  Hardware
Workers  Information

IT Infrastructure
Development Environment Architecture
Method Architecture  Tools Architecture  Enablement Architecture
Organization Architecture  Infrastructure Architecture  Adoption Architecture
An Architecture meets Stakeholder Needs

- Typical stakeholders of a development environment
  - Practitioner
    - Intuitive and correct behavior, performance, reliability, usability, availability, security
  - System administrator
    - Intuitive behavior, administration, tools to aid monitoring
  - Customer
    - Cost, return on investment, stability, schedule
  - Implementers
    - Clear requirements, simple and consistent design approach
  - Maintainer
    - Comprehensible, consistent and documented design approach, ease with which modifications can be made
  - Sponsor
    - Alignment of anticipated results with business and IT strategy
  - Strategic suppliers
    - Providing tools, training, infrastructure and second or third line support
An Architecture is Influenced by its Environment

- Method: Regulatory / organizational standards
- Tools: Existing “standard” tooling
- Training: An existing training curriculum
- Organization: Existing skills, organizational structures
- Infrastructure: Existing infrastructure
- Adoption: Approach to on-boarding teams on projects
Architecting addresses System Qualities

- Method: Usability, configurability, …
- Tools: Performance, scalability, …
- Training: Usability, …
- Organization: Evolvability, …
- Infrastructure: Performance, …
- Adoption: Return on investment, …
Agenda

- Start
- Development environment slides
- Architect slides
- Development environment architect slides

Finish
Summary

- Development Environment Architects are … Architects!
- Through analogy with a software architect, we can better-understand the role of the Development Environment Architect
  - Architecture
  - Architect
  - Architecting
  - Benefits of architecting
- This is an emerging domain that is becoming increasingly important!
Thank You