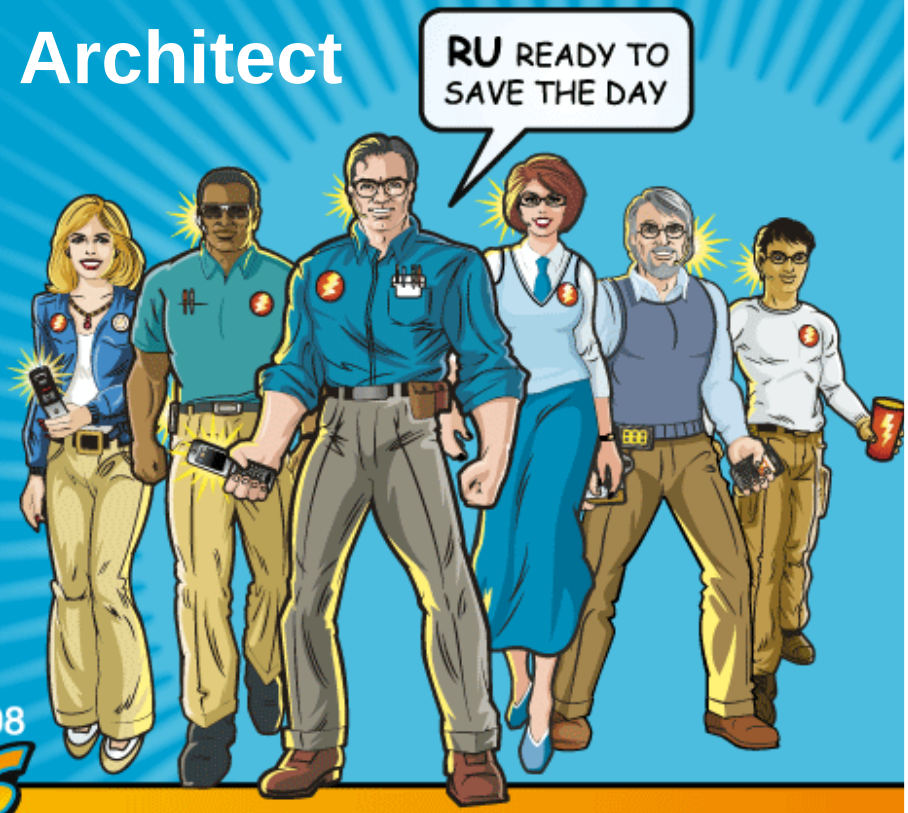
AM08

The Rise of the Development Environment Architect

Peter Eeles
Executive IT Architect, IBM
peter.eeles@uk.ibm.com



IBM Rational Software Development Conference 2008

WHERE TEAMS ARE R-HEROES

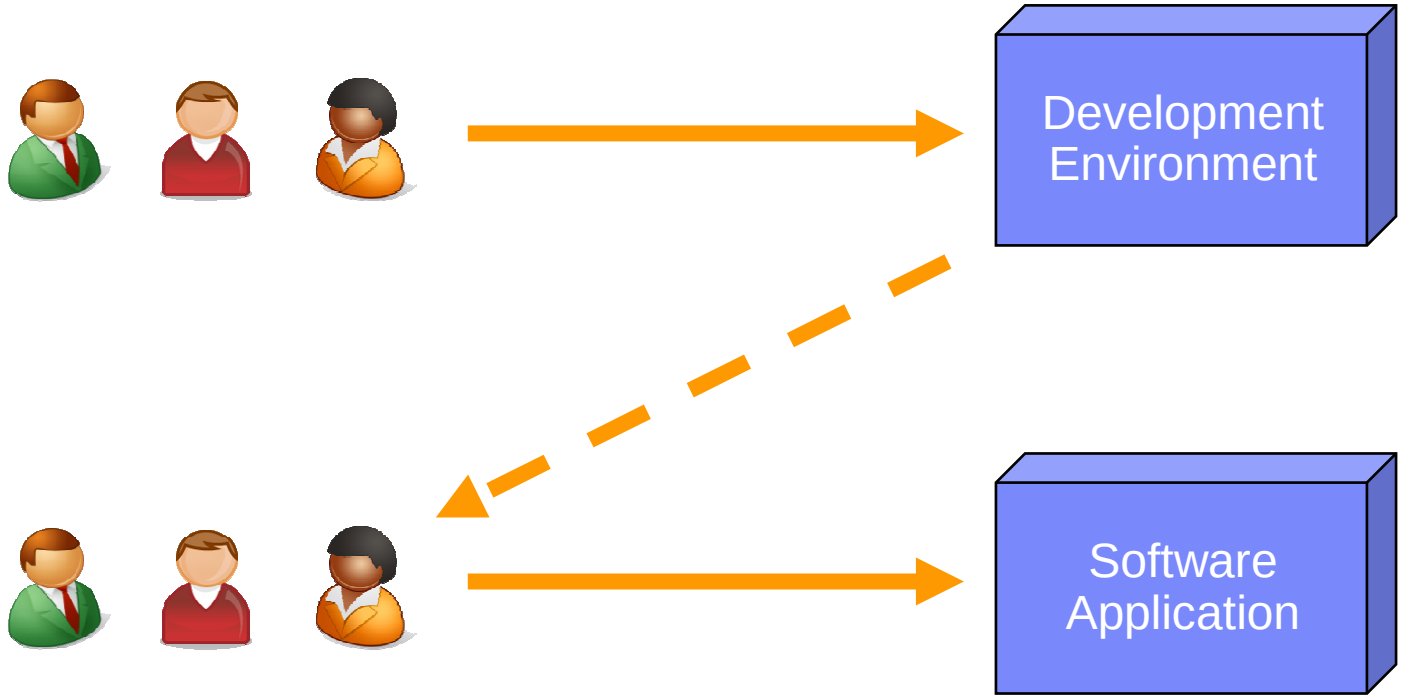
IBM

Rational. software

Agenda

- Introduction
  - The architecture of a development environment
  - Characteristics of a Development Environment Architect
  - The process of architecting a development environment
  - Benefits of architecting a development environment
  - The Development Environment Maturity Model (DEMM)
  - Summary

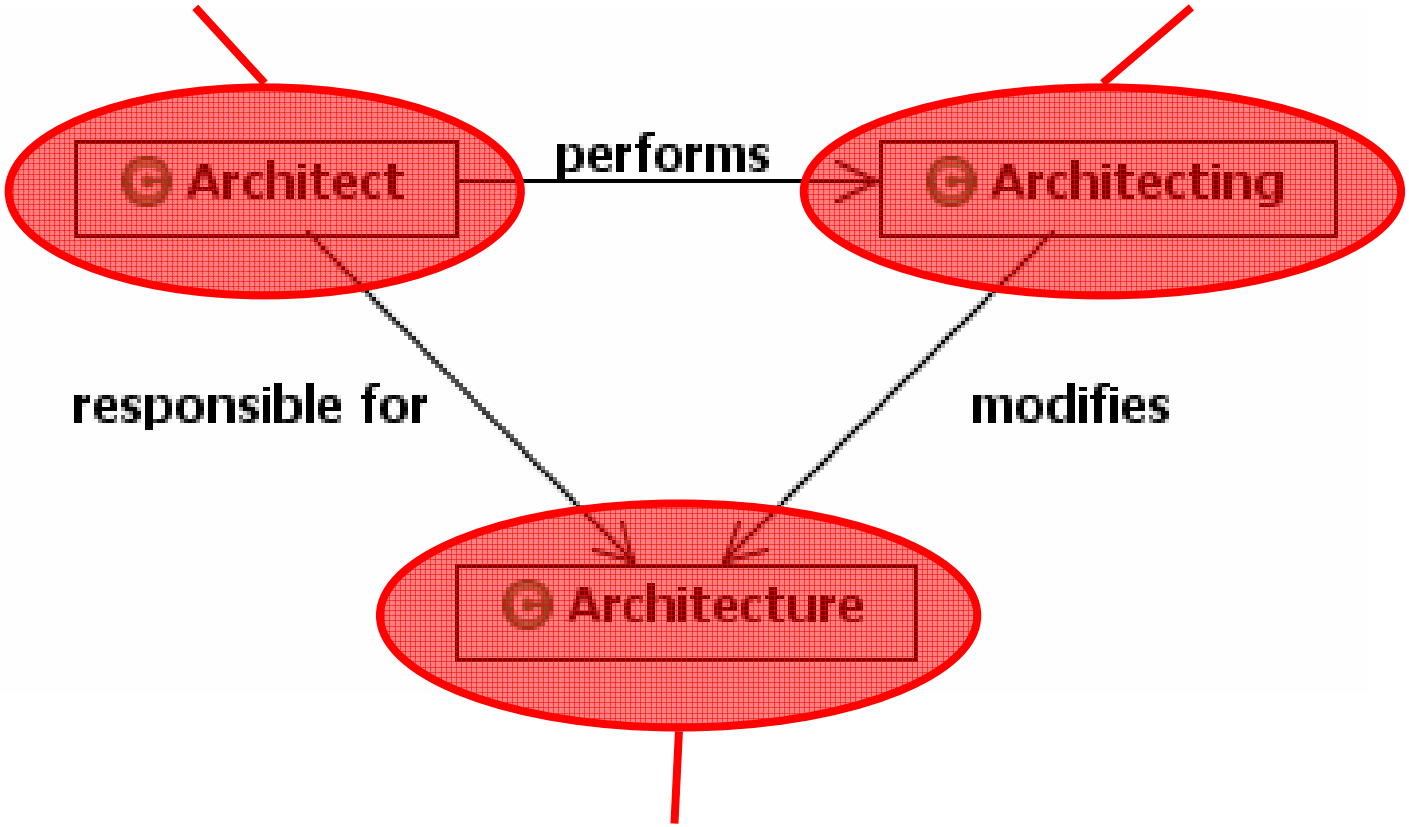
Architecting in context



Architecture, Architect, Architecting

Development Environment Architect

(Part of) a method

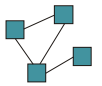


(Elements of) a development environment

Agenda

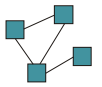
- Introduction
- The architecture of a development environment
- Characteristics of a Development Environment Architect
- The process of architecting a development environment
- Benefits of architecting a development environment
- The Development Environment Maturity Model (DEMM)
- Summary

Architecture



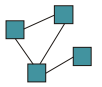
- *Architecture is the fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution. [IEEE 1471]*
- *The software architecture of a program or computing system is the structure or structures of the system, which comprise software elements, the externally visible properties of those elements, and the relationships among them. [Bass]*
- *[Architecture is] the organizational structure and associated behavior of a system. An architecture can be recursively decomposed into parts that interact through interfaces, relationships that connect parts, and constraints for assembling parts. Parts that interact through interfaces include classes, components and subsystems. [UML 1.5]*

An architecture defines structure



What are the key components of a development environment?

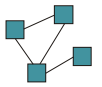
An architecture defines structure



	Element	Content	Deals With
Build	**Methods**	Process + method content Practices, standards, guidelines	Roles & responsibilities Work products Governance policies
Build	**Tools** (for automating aspects of the method)	Tool selection Tool configuration and integrations	Tool fitness-for-purpose Tool integrations Licensing
Build	**Enablement** (in methods and tools)	Training curriculum Training courses Mentoring materials	Training and mentoring
Build	**Infrastructure** (for methods, tools and enablement)	Infrastructure specification	Distribution of organization Solution packaging Installation and configuration
Build	**Organization** (to use and support methods, tools, enablement, infrastructure)	Organization specification	Skills Centers of excellence
Run	**Adoption** (of the environment)	Installation and configuration Training and mentoring Migration (including org. change)	Results Governance (e.g. measurement)

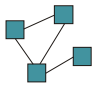
Functionality (E.g. change mgt, requirements mgt, testing)
Qualities (E.g. usability, scalability, reliability, cost)
Constraints (E.g. geographic distribution, migration)

An architecture defines behavior



- An architecture defines the interactions between structural elements
- These interactions provide the desired behavior
- Behavior of a development environment (examples)
  - Method — Guidance is provided when the method is followed
  - Tools — Work products are created when the tools are used
  - Enablement — Practitioners receive skills following training
  - Infrastructure — Qualities (such as tool performance) are realized
  - Organization — Assistance is provided by the helpdesk
  - Adoption — Projects are equipped with an appropriate environment

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
- Significant elements of a development environment (examples)
  - Method — Roles (and responsibilities), work products, tasks
  - Tools — Selected tools, integrations, licensing
  - Enablement — Training curriculum
  - Infrastructure — Geographic distribution, network connectivity
  - Organization — Organizational units (such as a helpdesk, communities)
  - Adoption — Rollout plan, migration strategy

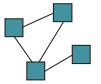
An architecture meets stakeholder needs

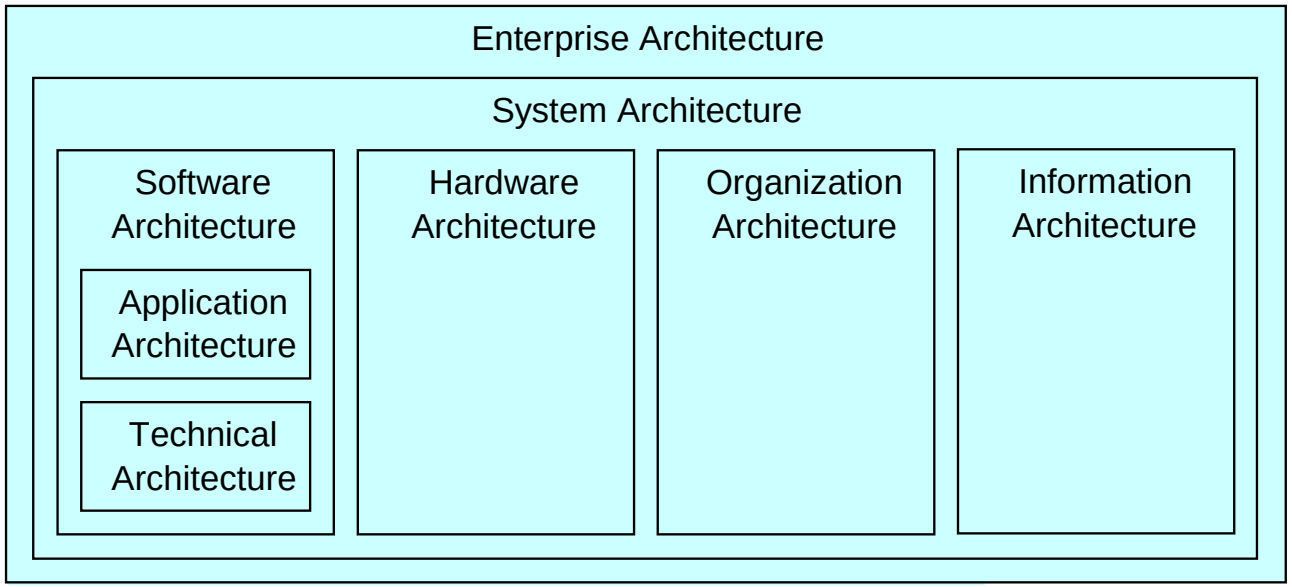
- Typical stakeholders of a development environment
  - Practitioner
    - Intuitive and correct behavior, performance, reliability, usability, availability, security
  - System administrator
    - Intuitive behavior, administration, tools to aid monitoring

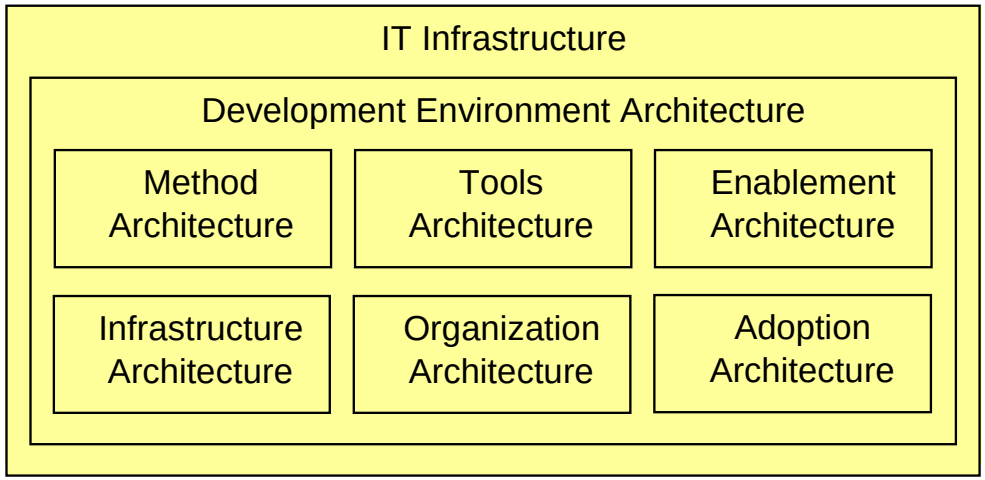
  - Customer
    - Cost, return on investment, stability, schedule
  - Implementers
    - Clear requirements, simple and consistent design approach
  - Maintainer

    - 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

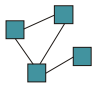
An architecture comes in many forms







Other characteristics of an architecture …



- An architecture is influenced by its environment
  - Method — Regulatory / organizational standards
  - Tools — Mandated tooling
  - Enablement — Existing elements of a training curriculum
  - Infrastructure — Existing infrastructure
  - Organization — Existing skills, organizational structures
  - Adoption — Ability to absorb change
- An architecture influences its environment
  - New skills, roles, method, tools, infrastructure, migrated data
- An architecture embodies decisions based on rationale
  - Decisions and rationale should be captured
- An architecture is present in every system
  - But is easier to communicate if it's documented

Agenda

- Introduction
- The architecture of a development environment
- Characteristics of a Development Environment Architect
- The process of architecting a development environment
- Benefits of architecting a development environment
- The Development Environment Maturity Model (DEMM)
- Summary

What are the key characteristics of an architect?

Characteristics of an architect

- The architect is a technical leader
  - We are the technical lead for the development environment
- The architect understands the development process
  - The architect must communicate with the team
- The architect has knowledge of the business domain
  - Our “business domain” is “software development”
- The architect has technology knowledge
  - The architect is not just a hand-waver
- The architect has design skills
  - The architect can architect ☺
- The architect has programming skills
  - The architect can “go deep” where necessary

Characteristics of an architect (2)

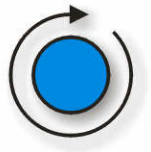
- The architect is a good communicator
  - As the technical lead, they must effectively interact with the team
- The architect is a mentor
  - And should grow capability
- The architect is aware of organizational politics
  - And needs to be “thick-skinned”
- The architect is a negotiator
  - The “usual” tradeoffs of functionality, quality, schedule and cost apply
- The architect role may be fulfilled by a team
  - Although there is normally a “lead architect”
- The architect makes decisions
  - Otherwise the team will make their own decisions

> *“The life of a software architect is a long and rapid succession of suboptimal design decisions taken partly in the dark.”* [Kruchten]

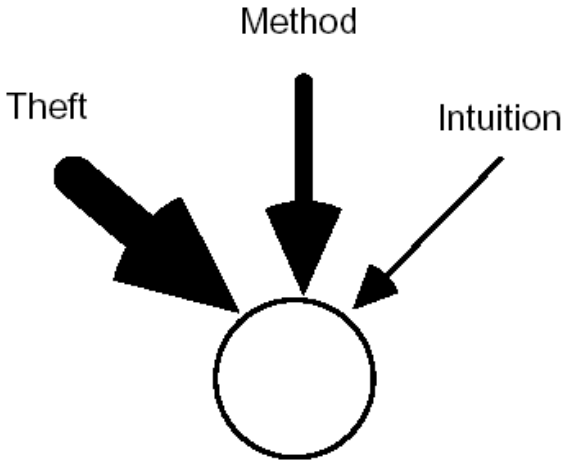
Agenda

- Introduction
- The architecture of a development environment
- Characteristics of a Development Environment Architect
- The process of architecting a development environment
- Benefits of architecting a development environment
- The Development Environment Maturity Model (DEMM)
- Summary

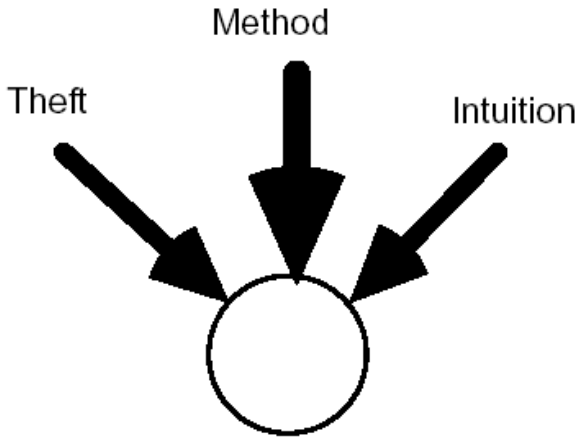
Characteristics of architecting



- 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



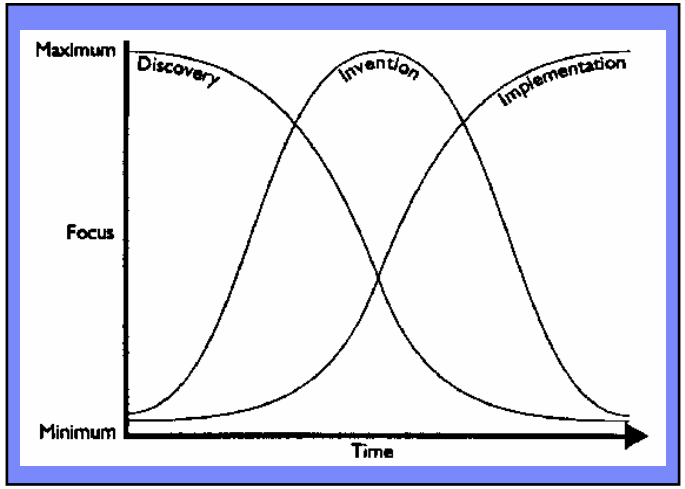
A classical system

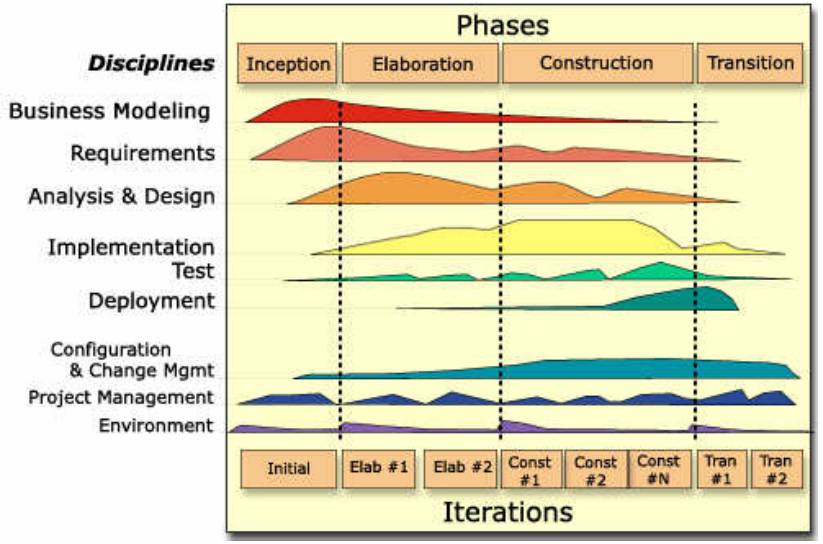


An unprecedented system

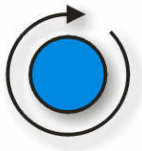
Characteristics of architecting (2)

- Architecting changes emphasis over time
  - Inception
    - Scope the problem, identify risks, define overall plan
  - Elaboration
    - Baseline the architecture
  - Construction
    - Complete the system
  - Transition
    - Transition to end users
- Architecting spans many disciplines
  - There is an involvement in requirements
  - There is an involvement in testing
  - There is an involvement in planning





Characteristics of architecting (3)



- Architecting involves many stakeholders
  - Understand who they are
- Architecting is involved in tradeoffs
  - Help stakeholders understand the tradeoffs
- Architecting considers reusable assets
  - Don’t reinvent the wheel
- Architecting is both top-down and bottom-up
  - Driven by requirements, constraints, lessons learned

Agenda

- Introduction
- The architecture of a development environment
- Characteristics of a Development Environment Architect
- The process of architecting a development environment
- Benefits of architecting a development environment
- The Development Environment Maturity Model (DEMM)
- Summary

What are the key benefits of architecting?

The benefits of architecting

- Architecting addresses system qualities
  - Method — Usability, configurability, …
  - Tools — Performance, scalability, …
  - Enablement — Usability, …
  - Infrastructure — Performance, …
  - Organization — Evolvability, …
  - Adoption — Installability, maintainability, …
- Architecting ensures architectural integrity
  - Across method, tools, enablement, infrastructure, organization, adoption
- Architecting supports the planning process
  - Provides input on dependencies, risks, effort, skills etc.
- Architecting provides a basis for reuse
  - Solution elements are often applied to more than one project

The benefits of architecting (2)

- Architecting drives consensus
  - By ensuring a fit between requirements and reality (the solution)
- Architecting helps manage complexity
  - By focusing on appropriate abstractions
- Architecting supports impact analysis
  - Through an understanding of the relationships between architectural elements
- Architecting reduces maintenance costs
  - Since “maintainability” is one of our considerations

Agenda

- Introduction
- The architecture of a development environment
- Characteristics of a Development Environment Architect
- The process of architecting a development environment
- Benefits of architecting a development environment
- The Development Environment Maturity Model (DEMM)
- Summary

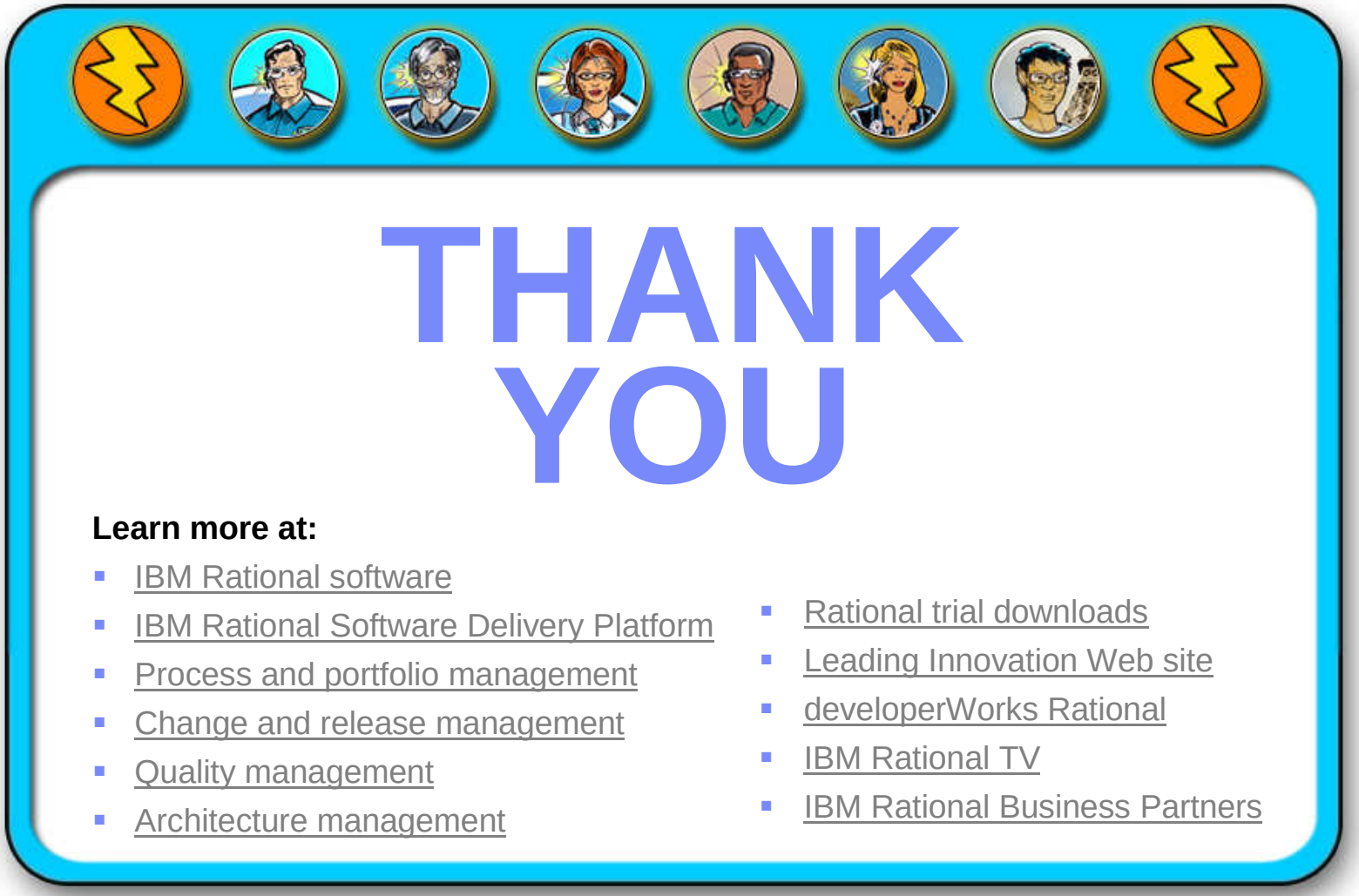
Development Environment Maturity Model (DEMM)

	Element	Level 1 *(Initial)* Ad hoc	Level 2 *(Repeatable)* Standard	Level 3 *(Defined)* Integrated	Level 4 *(Managed)* Measured	Level 5 *(Optimizing)* Improving
Build	**Methods**	Ad hoc	Controlled Now	Integrated End 08 → ?	Measured	Improving
Build	**Tools** (for automating aspects of the method)	Standalone Now	Standard End 08	Integrated → ?	Improving	Tuned
Build	**Enablement** (in methods and tools)	Inconsistent Now End 08	Consistent	Program → ?	Improving	Focused
Build	**Infrastructure** (for methods, tools and enablement)	Non-standard Now End 08	Standard	Single → ?	Monitored	Tuned
Build	**Organization** (to use and support methods, tools, enablement, infrastructure)	Inconsistent Now End 08	Consistent	Team Working → ?	Measured	Performing
Run	**Adoption** (of the environment)	Chaotic Now End 08	Good enough	Smooth → ?	Monitored	Streamlined

Summary

- Development Environment Architects are … Architects!
- Through analogy with software and systems engineering, we can better-understand the role of the Development Environment Architect
  - Architecture
  - Architect
  - Architecting
  - Benefits of architecting
- The Development Environment Maturity Model can help us better-understand where we are … and where we want to be
- White paper
  - http://www.ibm.com/developerworks/rational/library/edge/08/apr08/eeles/index.html


QUESTIONS



THANK YOU

Learn more at:

- IBM Rational software
- IBM Rational Software Delivery Platform
- Process and portfolio management
- Change and release management
- Quality management
- Architecture management
- Rational trial downloads
- Leading Innovation Web site
- developerWorks Rational
- IBM Rational TV
- IBM Rational Business Partners