KDM Analytics™

Software Assurance Ecosystem

Djenana Campara

Chief Executive Officer, KDM Analytics

Board Director, Object Management Group (OMG)

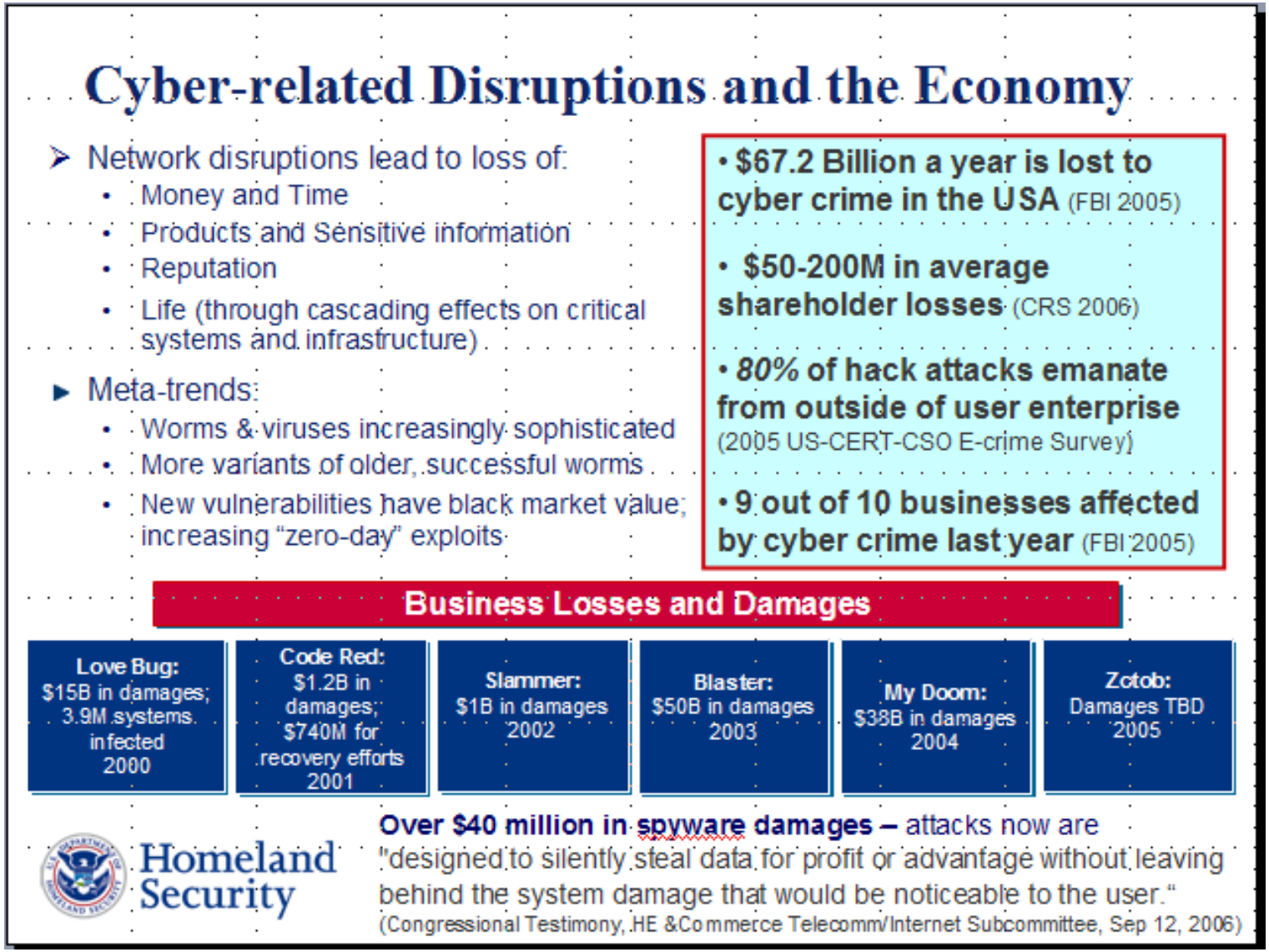
Co-Chair Software Assurance and Architecture Driven Modernization, OMG

Agenda

- Software Assurance (SwA)
  - The need
  - Definition
  - Current state of the SwA industry
- SwA Ecosystem
  - Realizing SwA Greatly Depends on Availability of Enabling Technologies
  - Introduction to SwA Enabling Technologies
  - ISO/OMG Tooling Standards Play Key Role in Forming Software Assurance Ecosystem
  - Detailed View of Infrastructure Software and Standards with Tools
- Software Assurance Ecosystem in Action

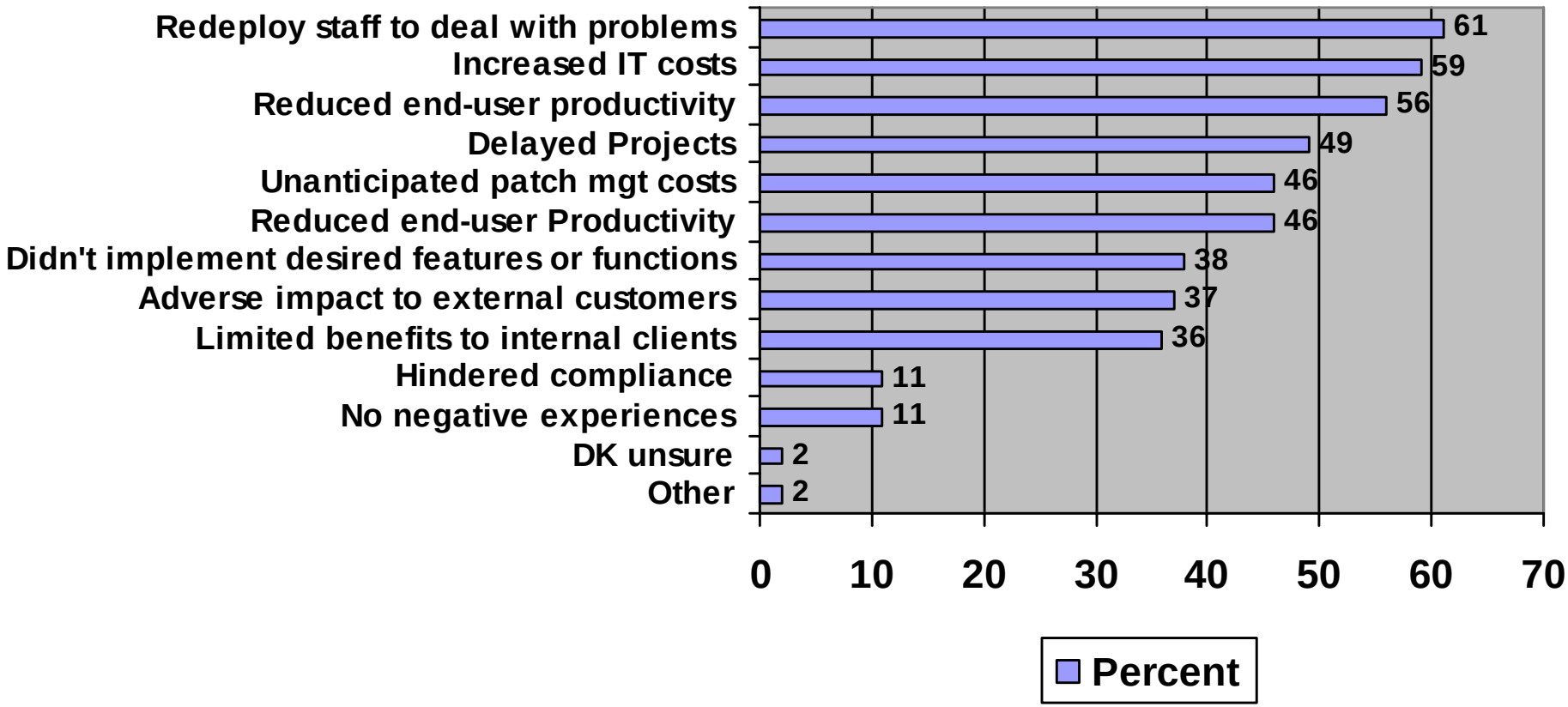
The Software Assurance

We Have a Problem!



IMPACT ON THE BUSINESS

of software flaws, vulnerabilities & malicious code



Survey conducted September 12-24, 2006

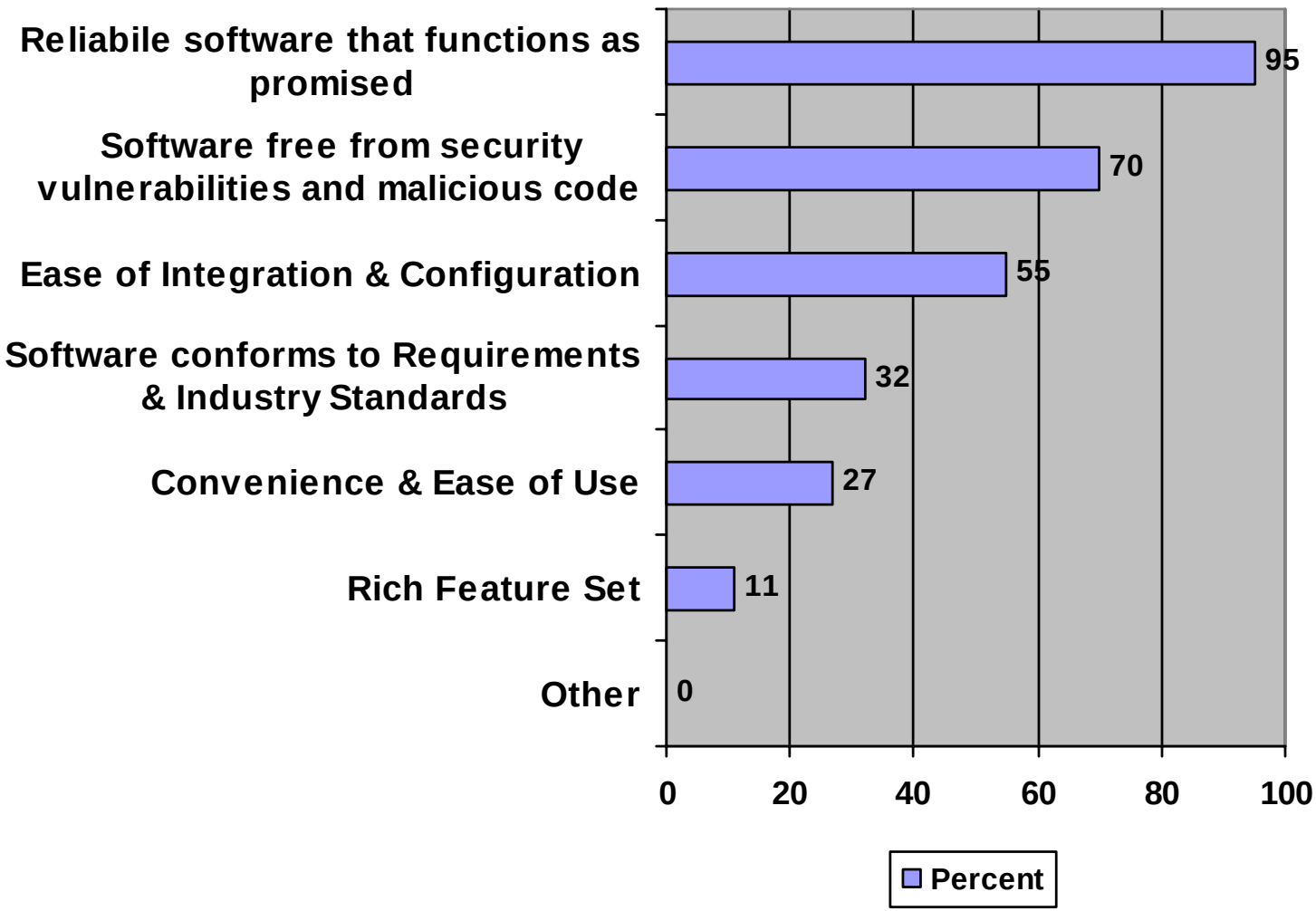
Respondents: 84 CIO Executive Council members & deputy members

CIO Executive Council

The Professional Organization for CIOs

KDM Analytics ™

MOST IMPORTANT ATTRIBUTES



CIO Executive Council

The Professional Organization for CIOs

KDM Analytics ™

SUMMARY from CIO Survey

- **Reliable software & vulnerability-free software are the top priorities;**
- **CIOs have low-medium confidence in software's ability to be free of flaws, security vulnerabilities and malicious code flaws;**
- **Eighty-six percent (86%) of CIOs rate the fundamental security of software as vulnerable or extremely vulnerable;**
- **The majority have had to redeploy staff, incur increased IT costs and suffer reduced productivity due to software flaws;**
- **Internal testing, contracts/SLAs and reputation among peers are the most preferable means for CIOs to determine if software is free of flaws;**
- **The majority would like vendors to certify software meets a designated security target and to scan for flaws and security vulnerabilities using qualified tools**

CIO Executive Council
The Professional Organization for CIOs

Software Assurance *– Introduced as a Solution*

- The level of confidence that software functions as intended and is free of vulnerabilities, either intentionally or unintentionally designed or inserted as part of the software”
  - [National Defense Industrial Association -NDIA].

- The justifiable trustworthiness in meeting established business and security objectives”
  - [Object Management Group - OMG]

Comes together as a formal framework for analysis and exchange of information related to software security and trustworthiness

Why Software Assurance is Critical

- Dramatic increase in mission risk due to increasing:
  - Software dependence and system interdependence (weakest link syndrome)
    - One vulnerability can cause a pandemic throughout the system(s)
  - Software Size & Complexity (obscures intent and precludes exhaustive test)
  - Outsourcing and use of un-vetted software supply chain (COTS & custom)
  - Attack sophistication (easing exploitation)
  - Reuse (unintended consequences increasing number of vulnerable targets)
  - Number of vulnerabilities & incidents with threats targeting software
  - Risk of Asymmetric Attack and Threats
- Software and the processes for acquiring and developing software represent a material weakness

The U.S. Government wants dramatic improvements in Software Assurance – both in industry and in government

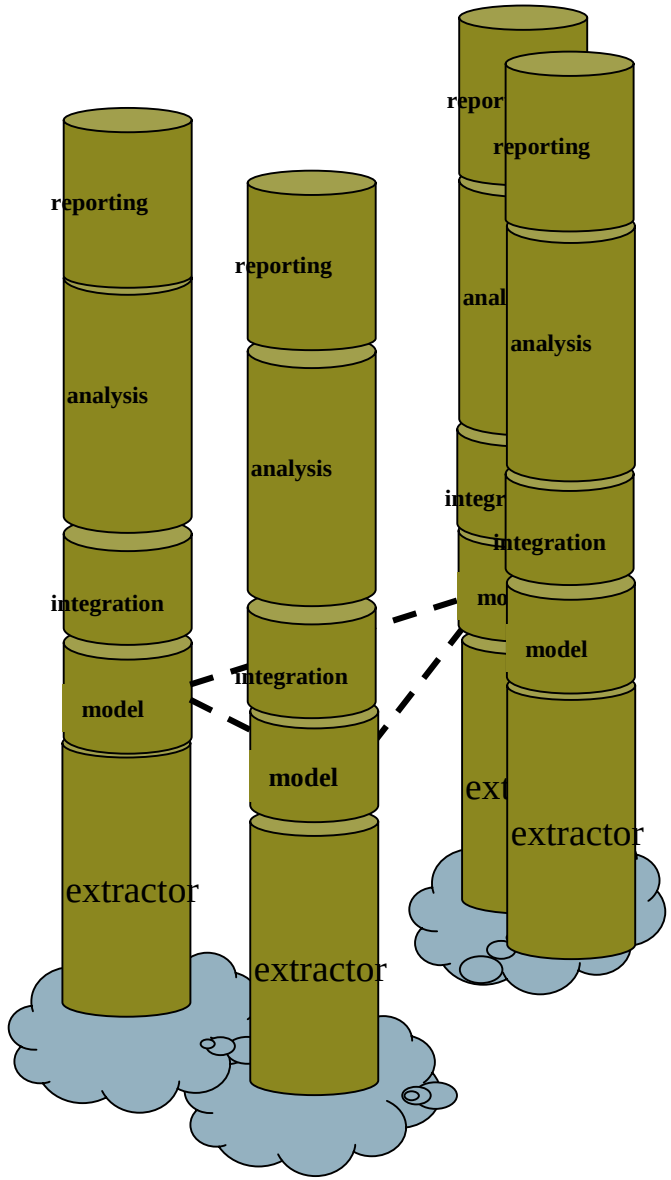
Current State of the Software Assurance Industry

- As software organizations recognize the need to evolve their systems beyond homogeneous and monolithic solutions to ones with a netted approach to architectures supporting COTS, multi-operating environments and multiple languages, security has become a significant challenge

- Simultaneously, the tooling industry which provides enabling technologies to build secure software systems has not keep pace with the software system evolution
  - a large gap has been created in which point tools have not kept pace nor provide the comprehensive evaluation required for a significant portion of the software industry, causing even greater security exposure

- Widening the gap further, software communities, in an effort to increase system security, focused their efforts on the development/compliance to more standards and guidelines on software systems, ignoring the evolution of technologies that can enable organizations to cost effectively implement and certify against said standards causing
  - More standards but less standardization
  - Less standardization leading to less confidence that products are trustworthy
  - Less confidence leading to lower levels of assurance

Disparate Tools



Traditional tools

Language & platform footprint

The Software Assurance Ecosystem

Realizing Software Assurance – Realizing what we already have

- Realizing Software Assurance is about enabling industry and government to **leverage** and **connect** existing standards, policies, practices, processes and tools, in an affordable and efficient manner

- The key enabler is the Software Assurance (SwA) Ecosystem infrastructure, which is an open standard-based integrated tooling environment that dramatically reduces the cost of software assurance activities
  - It integrates 3 different communities: Formal Methods, Reverse Engineering and Static Analysis to complete SwA enabling technology solution
  - Enables different tool types to interoperate
  - Besides expanding market for known players, it introduces many new vendors to ecosystem because they each have parts of the tool chain (and may not realize it). Eg. IBM, Telelogic, ASG, Relativity, Cognos)

- Many OMG/ISO standards play a vital role in the definition of the SwA Ecosystem infrastructure

The Software Assurance Ecosystem - Introduction

- The ecosystem provides a technical environment where formalized claims, arguments and evidence can be brought together with formalized and abstracted software system representations to support high automation and high fidelity analysis.
- It is based completely on ISO/OMG Open Standards to bring together a community of tool vendors that together will provide best in class solutions in the software assurance and modernization space.
- Architected with a focus on providing fundamental improvements in analysis through the direct support of analysis scoping and multi-pass analysis
- Three key open standards:
  - Semantics of Business Vocabulary and Rules (SBVR)
  - Software Assurance Meta-model (SAM)
  - Knowledge Discovery Meta-model (KDM)

Software Assurance Related ISO/OMG Standards: Current Status

Existing OMG Standards adopted by ISO

- ISO/IEC 19502 (MOF) / OMG MOF
- ISO/IEC 19501 (UML) / OMG UML2
- ISO/IEC 19503 (XMI) / OMG XMI

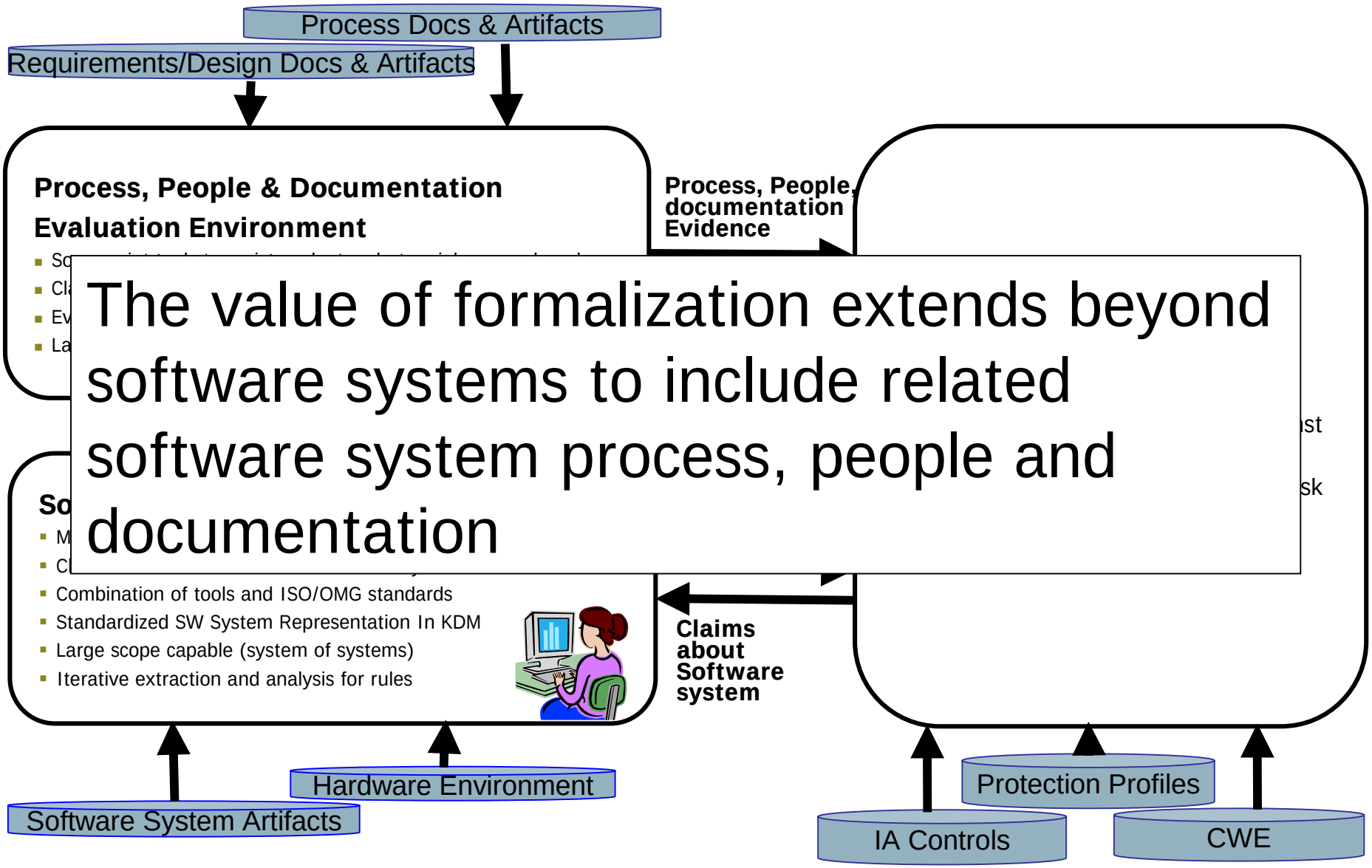
Existing OMG Standards proceeding with ISO Adoption

- OMG ADM Knowledge Discovery Meta-model (KDM)
- OMG Semantics of Business Vocabulary and Rules (SBVR)
- OMG Common Warehouse Meta-model (CWM)

Work in-progress OMG standards (will proceed with ISO adoption)

- OMG Software Assurance Meta-model
- OMG ADM Software Metrics Meta-model
- OMG Information Management Meta-model (IMM)
- OMG Business Process Meta-model (BPM)

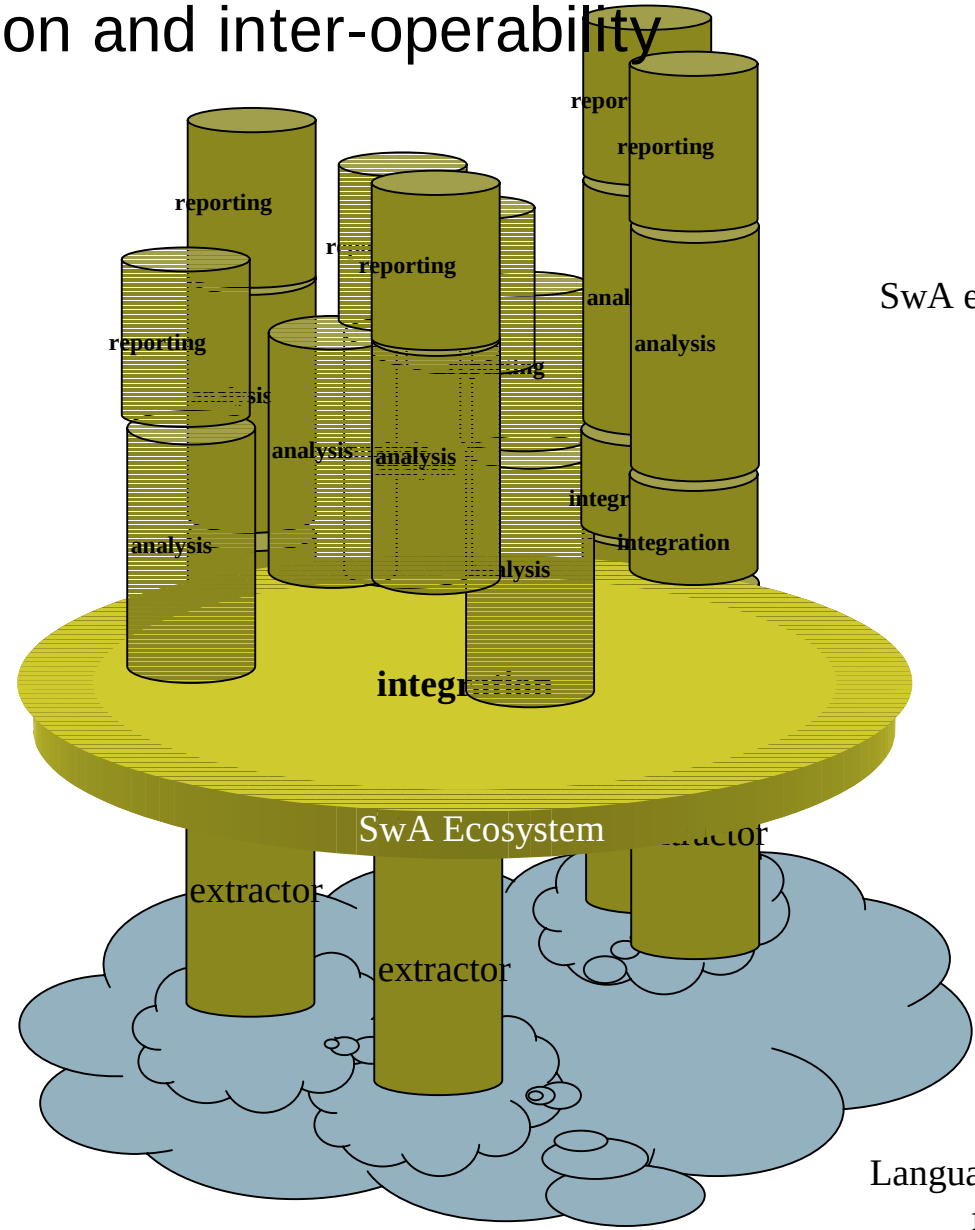
Software Assurance Ecosystem: *The Formal Framework*



Values

- Formal Representation of Claims, Arguments and Evidence
  - Provides precise and unambiguous specification
    - Legal and contractual agreements
    - Can generate documents and reports
    - Excellent communications vehicle between client and evaluator
  - Builds a repository of Claims and Arguments (reusable)
  - Improves objectiveness, accuracy of evidence collection through process, people, documents
- Highly automated collection of Evidence in relation to direct claims against software systems
- Software Assurance Repository brings all Evidence together with the Claims
  - Improves evaluations
    - Evaluations are consistent, objective and with repeatable results across different labs
    - Can easily and quickly look across all evidence to build a correlation model as continuous improvement of assessments
    - Automated validation of claims against evidence based on arguments
  - Highly automated and improved risk assessments using transitive inter-evidence point relationships

SwA Ecosystem Infrastructure: Creating Affordability through integration and inter-operability



SwA ecosystem tools

Traditional tools

Language & platform footprint

Software Assurance Ecosystem Main Parts

- Semantics of Business Vocabulary and Rules (SBVR)
  - Language for expressing Claims, Arguments and Evidence
  - Excellent representation of requirements in an easy to understand but formal manner
  - Precise and un-ambiguous
  - Used for both software system requirements and for software process, people and documentation requirements (eg. Completely covers IA Controls and Protection Profiles
  - Defining SwA vocabulary within SBVR framework – work part of the rollout plan

Software Assurance Ecosystem Main Parts (Cont.)

- Software Assurance Meta-model (SAM)
  - Work in progress at OMG
  - A repository structure for representing and exchanging Claims, Arguments and Evidence
  - Claims, Arguments and Evidence are documented using SBVR
  - A framework for building SwA related tooling for:
    - Improving repeatability and objectiveness of evaluations by automatically connecting evidence and claims
    - Greatly improving risk assessments through evidence correlation
    - Managing claims for consistency, understanding gaps, duplication, etc.

Software Assurance Ecosystem:
KDM: The KEY Ecosystem Standard
This is where it all comes together…

- Knowledge Discovery Meta-model (KDM)
  - A framework for documenting and formally representing existing software assets and their operational environment
    - Abstracted from source code for language independence

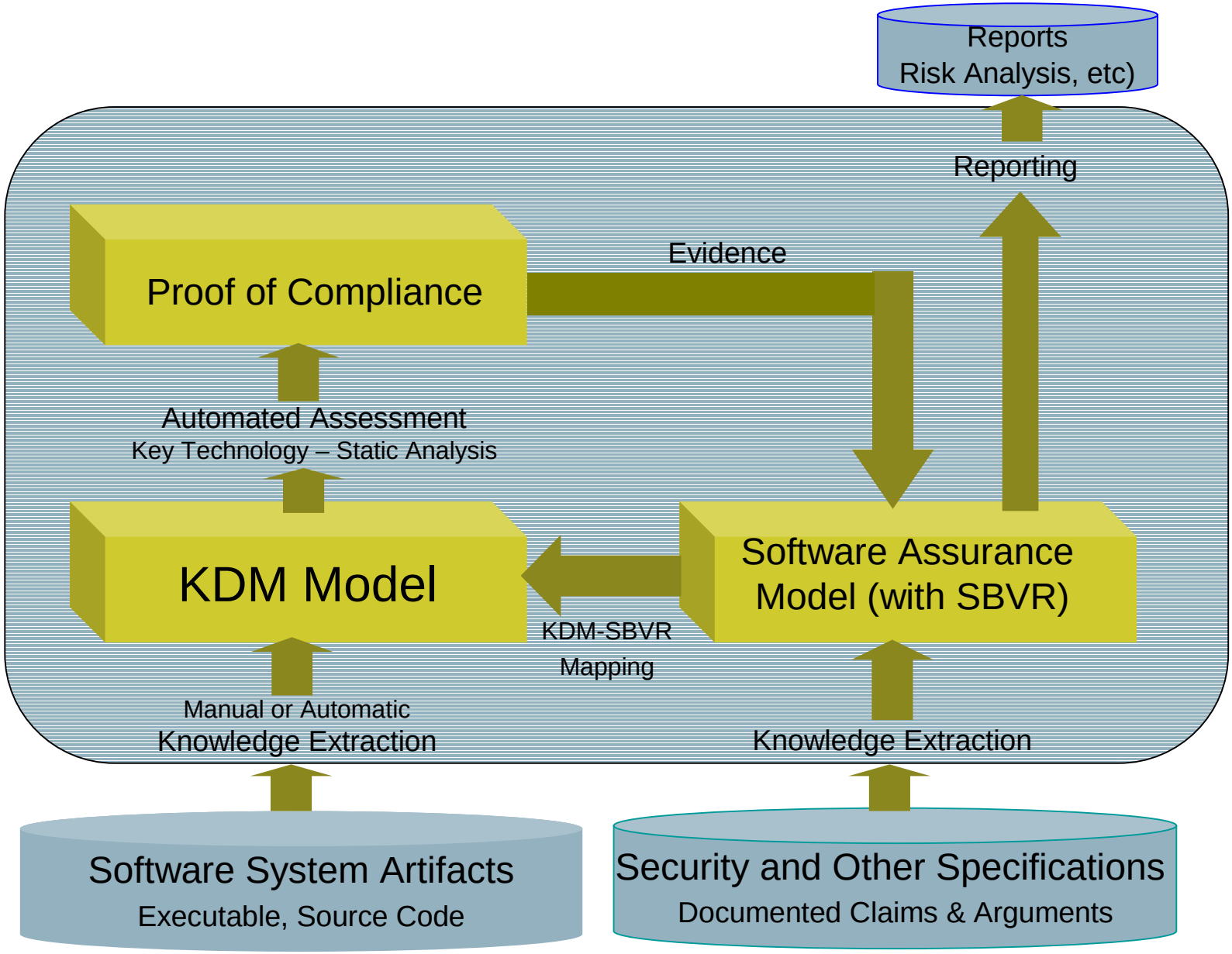
KDM is the key building block for the next generation of modernization and software assurance tools

  - Supports export and import of data currently contained within individual tool models that represent existing software assets. This facilitates continuous interoperability between existing SwA tools

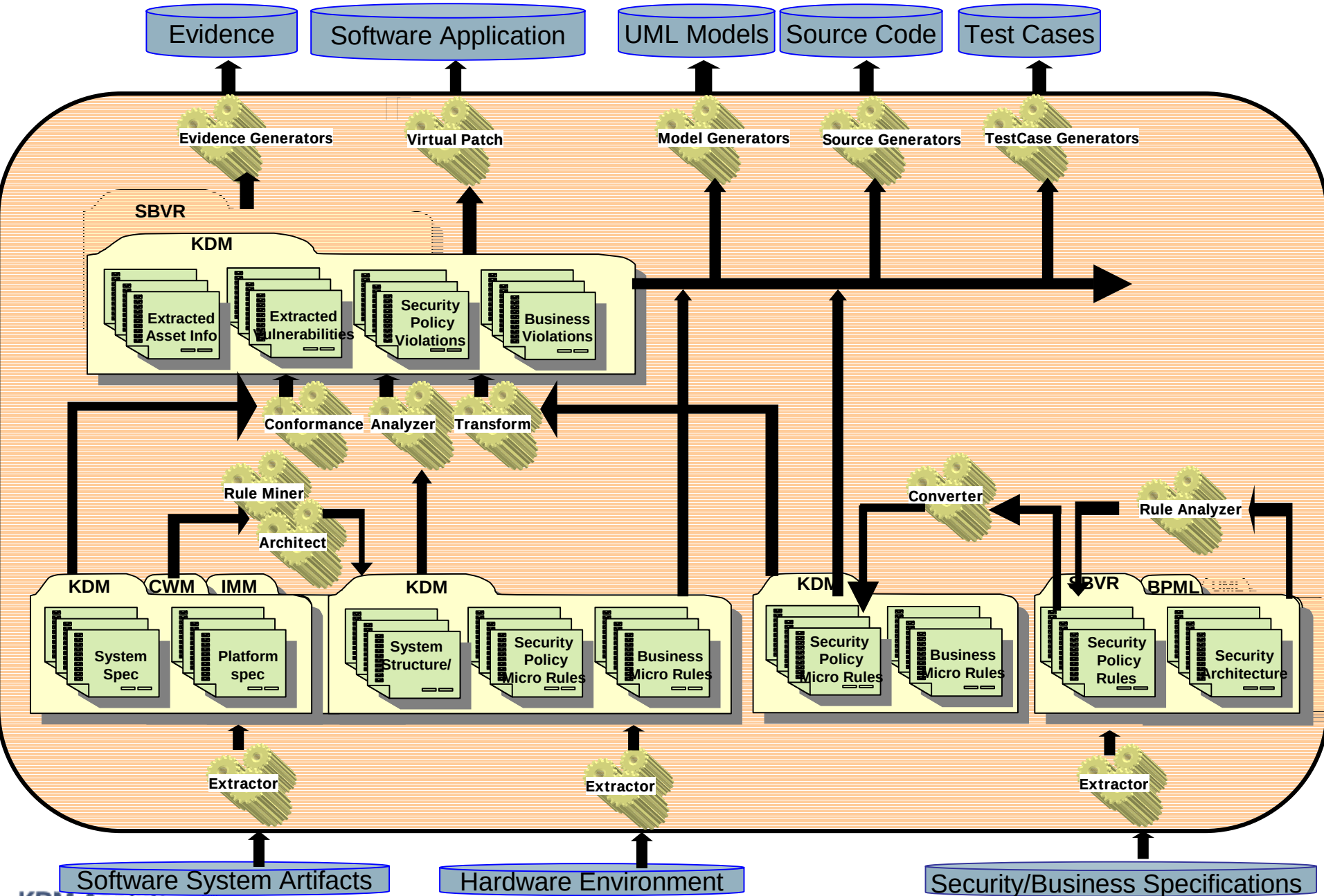
Why we care so much about Formalization

- Values
  - Unambiguous and Precise
  - Basis for legal and contractual agreements
  - Formalization enables automation and high fidelity analysis (accurate and transparent)
- Necessary to support automation of software assurance in KDM
  - Test case generator tools (CWE test case generator)
  - Static Analysis tools (CWE Analyzer, IA CtI/PP SW Requirements Compliance Analyzer)
  - Virtual Patching tools
  - Exploit generator tools
  - Penetration testing tools
- Why SBVR is the best formalization language
  - Business rule language designed for formalization of business and security requirements and is already an OMG standard.
  - Rules can be expressed in structured English
  - It's built upon OMG stack of standards that allows automatic generation of repository and inter-exchange (XMI)
  - Enables mapping between other OMG specifications (in our case KDM)

Software Assurance Ecosystem – Technical Implementation



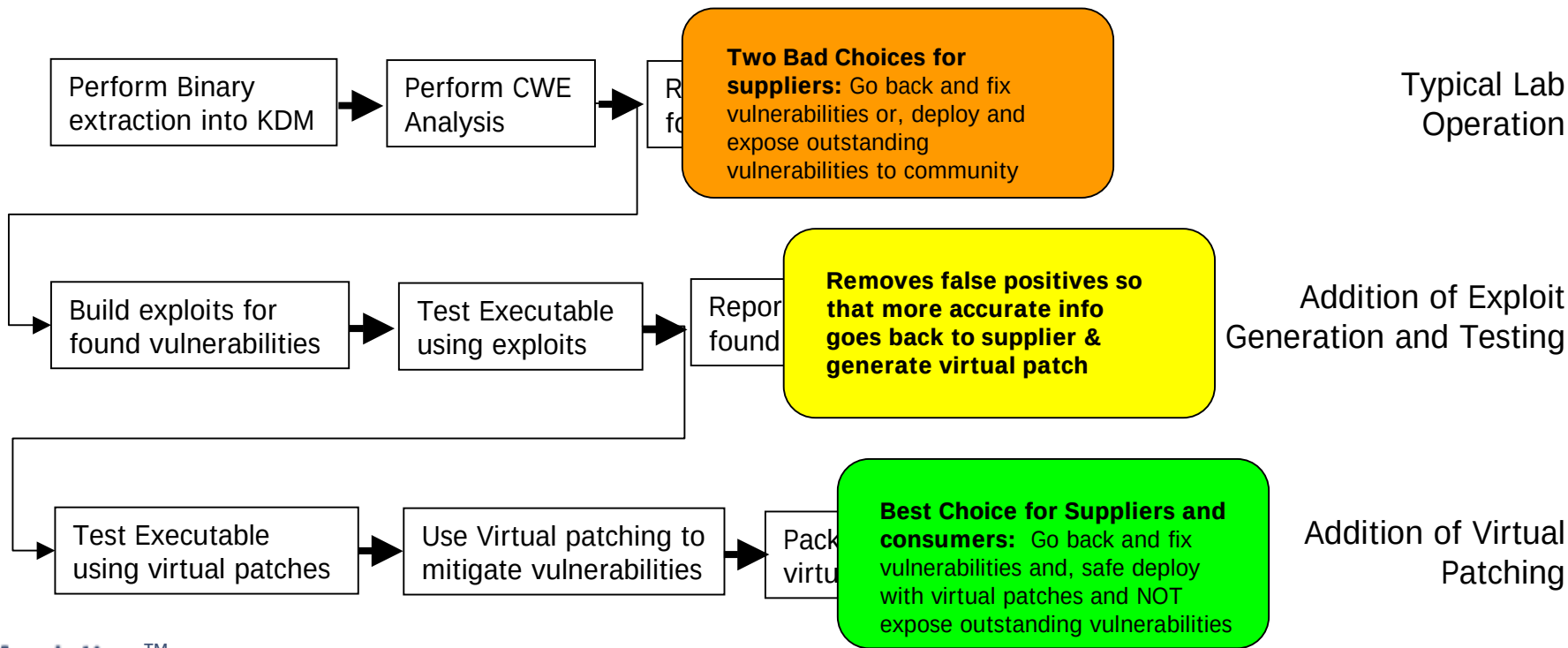
The Landscape - Infrastructure Software and Standards with Tools



The Software Assurance Ecosystem in Action

3rd Party Evaluation of Applications – LAB Environment

- The Open standard-based SwA ecosystem can be leveraged to increase deployability of tested applications. The following are workflow and steps:
  - use of software assurance tools to perform CWE-based analyzes of application
  - increase accuracy through building and applying exploit testing where weakness identified
  - provide virtual patches to mitigate effect of vulnerabilities
  - package application and virtual patch into deployable solution creating WIN-WIN situation for both supplier and consumer



Thank-you