Education

SMI-S: Building the Case for a Standard

Paul von Behren, Symantec Corporation

SNIA Legal Notice

- The material contained in this tutorial is copyrighted by the SNIA.
- Member companies and individuals may use this material in presentations and literature under the following conditions:
  - Any slide or slides used must be reproduced without modification
  - The SNIA must be acknowledged as source of any material used in the body of any document containing material from these presentations.
- This presentation is a project of the SNIA Education Committee.
- Neither the Author nor the Presenter is an attorney and nothing in this presentation is intended to be nor should be construed as legal advice or opinion. If you need legal advice or legal opinion please contact an attorney.
- The information presented herein represents the Author's personal opinion and current understanding of the issues involved. The Author, the Presenter, and the SNIA do not assume any responsibility or liability for damages arising out of any reliance on or use of this information.

  NO WARRANTIES, EXPRESS OR IMPLIED. USE AT YOUR OWN RISK.

Abstract

- SMI-S: Building the Case for a Standard
  - This tutorial describes how the SMI-S standard benefits four primary players in storage management solutions – end users, vendors of storage products, application (such as SRM) vendors, and integrators. Standards-based solutions help end-users avoid vendor lock-in, provide choices in storage management applications, and eliminating the cost of vendor-specific agent infrastructures. Vendors of storage products benefit by gaining application support with little or no vendor-specific development costs. Application vendors and integrators benefit by gaining storage support with little or no vendor-specific development costs. This tutorial also looks at innovative solutions that can exploit a standards-based infrastructure such as SMI-S – for example, Management Frameworks.
  - This tutorial also talks bout the past and future evolution of SMI-S; how the standard started by providing basic storage networking management tasks, adding more functionality and increasing the scope to meet the requirements of end-users and storage vendors, and plans for the future.

Overview

- Qualities of a "good" standard?
  - Examples of good and not-so-good standards
- How this applies to SMI-S
  - How SMI-S measures up to these qualities
  - How SMI-S stakeholders benefit from the standard
  - How the evolution of SMI-S increases the benefits

Check out SNIA Tutorial:

Solving Business-Oriented Goals with SMI-S

What makes a “good” standard?

- Vendor Neutral
- Well documented
- Sustained, democratic governance
- Completeness, with room for extensibility
- Conformance tests
- No more complex than necessary
- Improves efficiency for stakeholders
- Provides a platform for innovation

Example of a "good" standard:
Intermodal Containerized Shipping

- Created in 1956, has revolutionized shipping
  - Improved logistics and reduced costs
- Standard sized containers: 20', 40' and 45'
- Transportable on ship, rail, truck, etc.
- Arbitrary contents, including modular offices and hospitals
- Standard size helps to optimize design for warehouses, ships, terminal ports, handlers, etc

Example of a sub-optimal standard: Master Boot Record disk partitions

- De facto standard – allows various PC platforms to coexist as different partitions of disk
- Essential functionality for x86 system bootstrap
- No written specification, no owning organization
  - Occasional collisions when different stakeholders decide to use same value or area for different purposes
- Limited long-term view
  - Every couple years, some other limit (32 meg limit, 2 gig limit, 8 gig limit, cylinder related limits, …) surfaces
  - In general, tough to use newer disks with older motherboards
  - Complete dead end at 32-bit block address limit (2 TB)
- Well-designed replacement (GPT) tied to languishing BIOS replacement (EFI)
  - Over-complexity hampering acceptance

Informal and Formal standards

- 'De Facto' Standards
  - Informal – defined by one vendor or organization without standards credibility
  - Often created to solve a short-term problem without necessarily looking at long-term issues
  - Tend to provide limited interoperability
  - Tend to reduce competition
- 'De Jure' Standards
  - Created/recognized by an authorized body
  - Intended to increase competition and interoperability

Standards Lifecycle

- Standards that "stick" often start as de facto (informal), then mature into de jure (official) standards
- Initial work serves as prototype – helps work out the kinks
- Stakeholders recognize that industry-wide and open participation strengthens the standard
- Also recognize a need for an organization to oversee evolution of the standard

Example: Secure Socket Layer

- The "S" (security) in HTTPS
- Originally developed by Netscape for e-commerce,
  - Publicly shared specification, but not a 'de jure' standard
  - Developed quickly, became obligatory for e-commerce
- At the time, difficult to imagine an internet without Netscape, but some folks wanted more rigor
- Took SSL to IETF to become a de jure standard (RFC)
  - Renamed as Transport Layer Security (TLS)
  - Improved documentation
  - Governance
  - Vendor Neutral
  - Innovation – SSL/TLS now integrated into many solutions

How SMI-S Measures up

- Look at the characteristics of a good standard (completeness, conformance tests, well documented, …) and see how SMI-S supports the goal
- In particular, look at the benefits to key stakeholders
  - Storage vendors
  - Application (e.g., SRM) developers
  - Integrators
  - End users

Vendor Neutral

- SMI-S published as an ANSI (US) and ISO (international) available to any vendor
- SMI-S developed in SNIA Technical Working Groups (TWGs)
  - TWGs open to any SNIA member
- Conformance tests open to any vendor

SMI-S Documentation

- Each release first developed as a SNIA specification, selected releases submitted to ANSI, then ISO
- Documentation workload shared by volunteer authors supported by TWGs
- Managing this scope of documentation (SMI-S 1.1 has 1474 pages) has been challenging
  - Learning from ANSI, ISO, and peers (T10, T11)
  - SMI-S 1.2 broken into 9 separate books (aligned with topics/TWGs)
  - Content Author Guide for style, tool usage
  - Professional editors pull together the entire collection, help with standardization tasks

Building a case for a standard…

SMI-S Documentation

- With each release, documentation improves
  - Authors learn more about writing standards
    - Reading ISO style guide, incorporating suggestions
  - Addressing comments from ANSI/ISO reviewers
  - SNIA Review process improving
  - More reliance on professional editor to clean-up text and formating
  - Tool improvements – portion of each profile generated from machine-readable XML files
    - Verifies that programmable names are valid
    - Same files used by CTP – help eliminate inadvertant off-by-one differences between spec and tests

SMI-S Governance

- SNIA defines policies for TWGs and technical architectures (like SMI-S)
  - IP Policy
  - Vendor Neutrality
- SMI Technical Steering Group (TSG) oversees SMI TWGs
- Storage Management Initiative (SMI) formed 2006
  - Formed as SNIA "initiative"
  - Subsumed SMF and other SMI-S related non-technical entities
  - SMI Governing Board (and committees)
    - Schedules, budget, marketing, run plugfests

SMI-S Completeness, extensibility

- Early versions focused on basics (discovery, minimal provisioning) of commonly deployed SAN devices
  - Initial goals defined in 'functionality ladder' from Bluefin
  - All documented in SMI-S 1.0
- Later versions filling in gaps
  - Advanced and newer functionality (e.g., Thin Provisioning)
  - Optimization (e.g., bulk retrieval of common info from arrays)
  - Additional devices (e.g., Host Hardware RAID controllers)

SMI-S Extensibility

- CIM Common Information Model has served well in allowing us to add more devices, functionality
- SMI paving the way for vendor extensions
  - External to the standard, but format common with standard profiles
  - Created by vendors, then shared
  - Serve as incubators for new spec functionality
  - SMI-S 1.4.0 draft contains rules for keeping extensions compatible with existing profiles

SMI-S Conformance Tests

- CTP tests created for SMI-S 1.0 and extended with each follow-on release
  - Infrastructure tests
    - Independent of storage
    - Tests interoperability of the transport
  - Read-only test of information
    - Are mandatory elements provided?
    - Formatted as expected?
  - Active management tests
    - E.g., Create LUNs or Zones

Conformance Tests

- Over 500 SMI-S provider and client products have been tested through the Conformance Testing Program (CTP) for SMI-S 1.0/1.1
- CTP provider test development for SMI-S 1.2 wrapping up
  - Vendor testing starting
  - Large investment in updates to help "raise the bar" in CTP testing

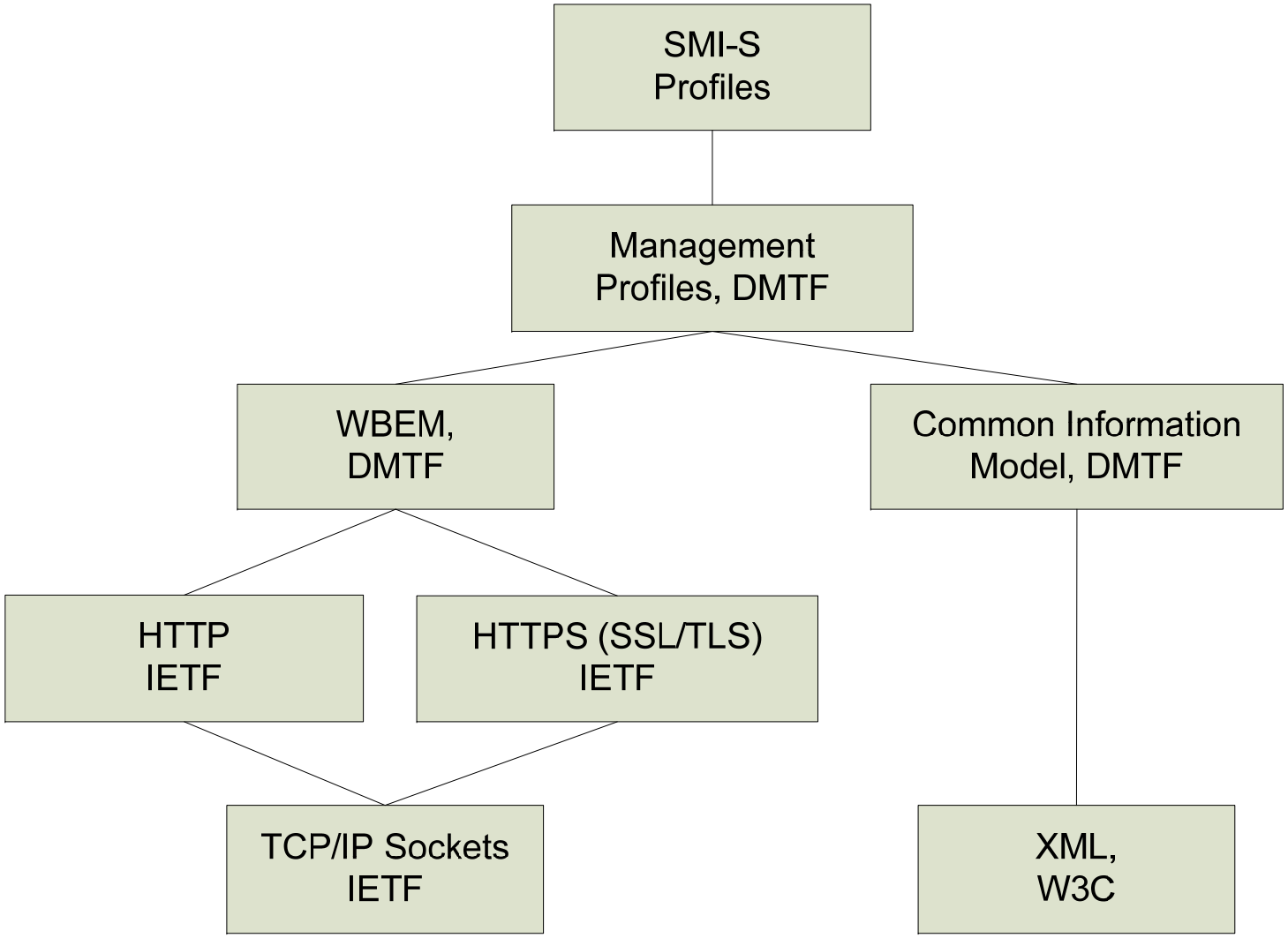
No more complex than necessary

- Cross-references are the mantra for standards like SMI-S
  - Build on and refer to other standards rather than re-inventing
  - SMI-S builds on standards from T10 (SCSI) T11 (FC), T10 (ATA), IETF (iSCSI, networking, security), DMTF (management), and other organizations
    - 39 standards referenced in SMI-S 1.2 Architecture book – these are standards with global relevance to SMI-S
    - Individual books reference other standards
- The result is that the bulk of SMI-S is about storage management

SMI relationship to other standards

- Peer relationships
  - SMI-S defines management standards for devices with datapath standards defined elsewhere (SCSI, FC, …)
  - SMI-S emphasizes use of names standardized elsewhere
    - Help end users correlate info from SMI-S and non-SMI-S applications
- Layering - building on standards
  - SMI-S builds on network-facing management standards from DMTF
    - SMI-S focuses on storage
    - SMI-S indirectly builds on network standards from IETF, XML from W3C
    - See next slide
  - SMI-S defines some profiles based on standard APIs (e.g., HBA API)
- Filling Gaps
  - SMI-S utilizes standards (e.g., SLP) to cover functionality not addressed elsewhere

SMI-S Building on other standards



Managing Complexity

- Early SMI-S focused on storage functionality incorporated in the standard
- Recent activities focused on production-readiness
  - How to install multiple-vendor SMI-S solutions
  - Trying to avoid adding additional complexity to SMI-S
  - Finding other vehicles
    - Provider portal
    - Best practices papers

SMI-S efficiency for stakeholders

- SMI-S benefits to storage vendors
  - Encourages development of third-party storage management applications
    - Offer customers multiple applications
  - Minimize need to develop or maintain in-house management infrastructure
- SMI-S benefits to application vendors
  - Single infrastructure/transport common to multiple vendors and device types

SMI-S efficiency for stakeholders

- SMI-S benefits to integrators
  - Common management platform for many vendors and devices
    - Minimize investment in multiple, proprietary infrastructures
  - Tools, trouble-shooting techniques, … common to multiple vendors and products
- SMI-S benefits to end users
  - Minimize vendor lock-in
  - Share benefits to integrators
  - Platform for innovation – future products

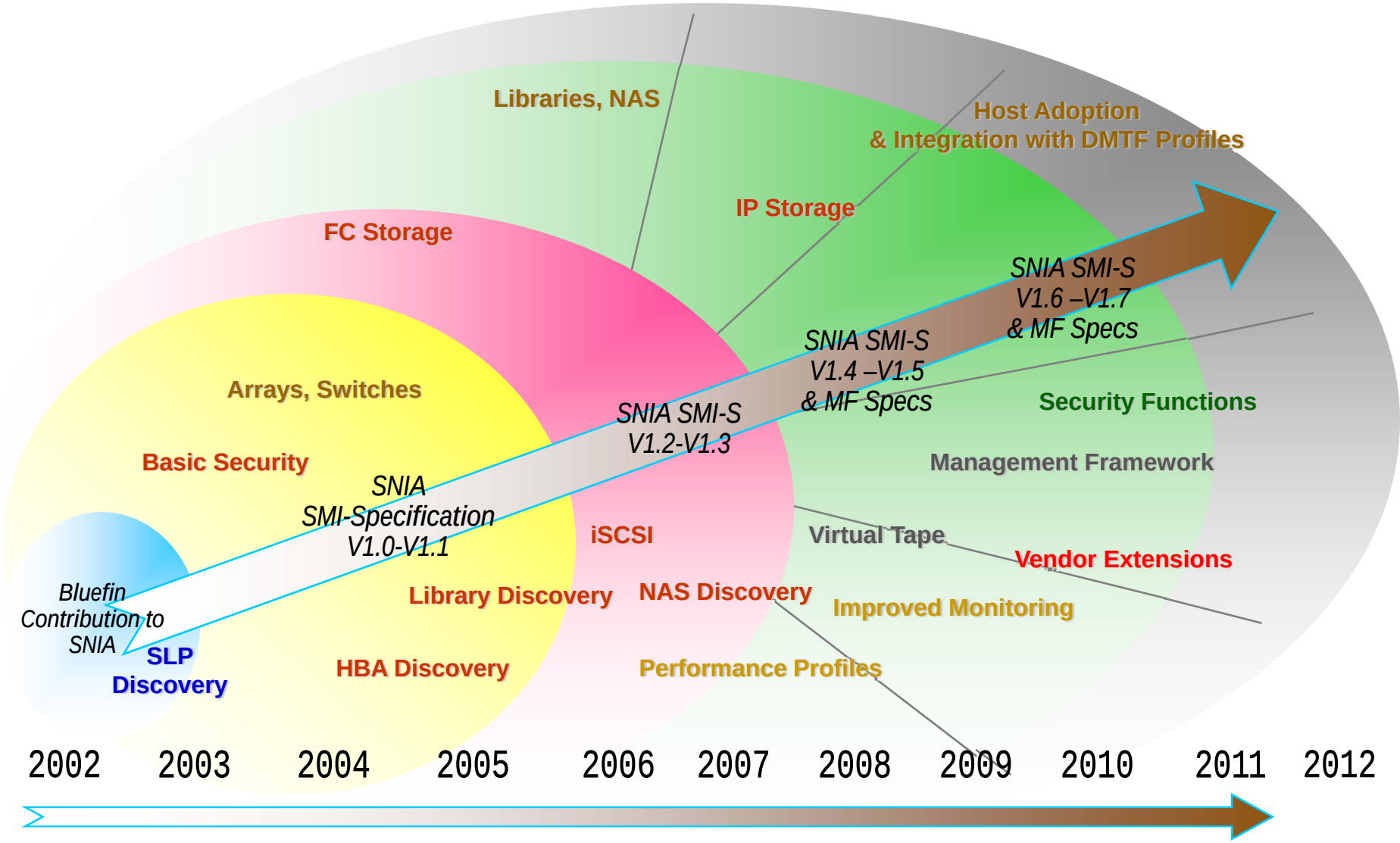
SMI-S: platform for innovation

- SMI-S developed with a goal of enabling innovation
  - Imagine other standards layered over SMI-S
    - For example, SNIA Management Frameworks
  - Exploit commonality across profiles
    - SMI-S Common Recipes show how an application might compose a SAN-wide topology using results from many SMI-S profiles
- Potential for Automation
  - Recovery when events are reported
  - Preventative maintenance
    - Monitor capacity growth and expand a LUN off-hours

SMI-S Roadmap

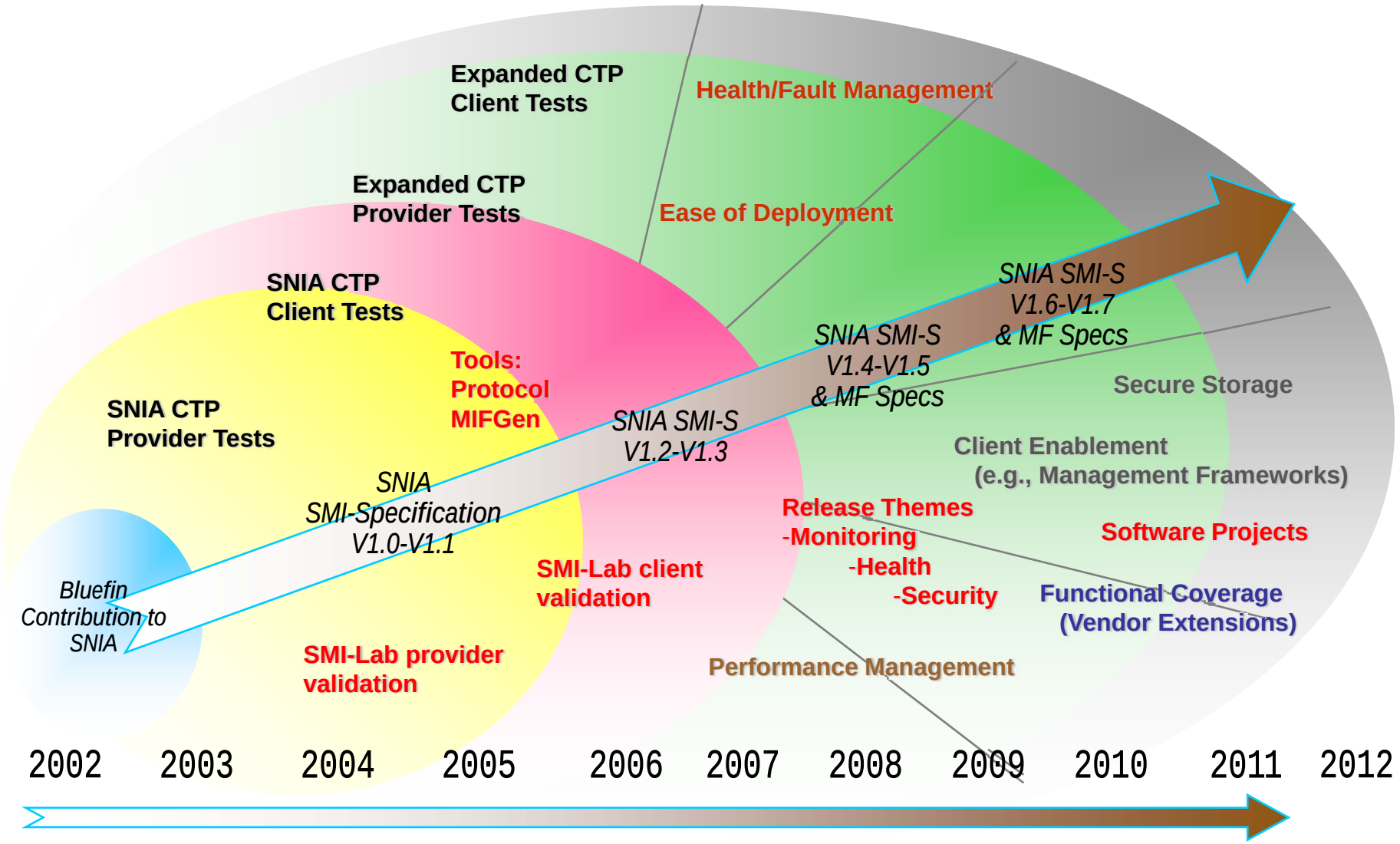
- Where are we now
  - SMI-S 1.1.1 in ANSI and ISO standardization processes
  - SMI-S 1.2 completed early 2008 – released as SNIA Architecture but not being submitted to ANSI/ISO
  - SMI-S 1.3
    - CTP testing in progress
  - SMI-S 1.4 and Management Frameworks
    - Content being developed
- The following slides show the SMI-S roadmap past and planned future

Technical Roadmap



Storage Management Initiative

Ecosystem Road Map – The journey continues



Q&A / Feedback

- Please send any questions or comments on this presentation to SNIA: trackvirtualization@snia.org

Many thanks to the following individuals for their contributions to this tutorial.

- SNIA Education Committee

Mike Walker
Tom Mancuso
Gary Phillips