

Polymorphic Protection Profiles

Presentation to the
11th International Common Criteria Conference
Antalya, Turkey
September 21, 2010

Rance DeLong
Consultant

Polymorphic Protection Profiles (PPPs)

- **Motivation:** Why are polymorphic protection profiles necessary?
- **Description:** What is a polymorphic protection profile?
- **Evaluation:** How can PPPs be evaluated?
- **Application:** How are PPPs used?

Motivation for Polymorphic PPs

Why are Polymorphic Protection Profiles necessary?

- We introduce MILS and the challenges it poses for standard CC PPs
- We consider the MILS Network System PP example

MILS* in a nutshell

- Business View
 - A **component-based approach** to security-critical systems (see: Ref. [4])
 - A **global commercial marketplace** of high-assurance components
 - A cost-effective **development and certification strategy** for systems
 - Goals: Functionality, Security, Certifiability, Affordability
- Technical View
 - An approach to system **decomposition** (MILS policy architecture)
 - A resource-sharing **implementation technology** (MILS components)
 - High-assurance components predefined by **MILS protection profiles**, e.g.,
 - MILS Network System PP, MILS Console System PP, MILS File System PP
 - MILS component integration **theoretical foundations and standards**
 - Foundation, tools and standards for **compositional evaluation and certification** (see: 8th ICCC [1])
 - Ability to predict properties of MILS systems based on those of components

* MILS was originally an acronym for “Multiple Independent Levels of Security”. We now use it as a proper name.

MILS Network System (MNS)

- A class of MILS subsystem
 - Provides network services within a MILS architecture
 - Some instances intended for environments requiring high assurance
- Complex System with a Range of Possible Features
 - Protocols and Services - e.g., TCP, IP, UDP, SNMP, BGP, DNS, network time, ...
 - Functionality equivalent with commercial IPv4 and IPv6 implementations
 - Interoperability with other “closed” network standards, e.g., CIPSO, CALIPSO*
- Range of Functionality × Assurance Strategies
 - Need high-assurance, not just another “low-assurance” offering, however ...
 - Cannot provide all functionality (e.g., all of IPv6) with highest level of assurance
 - Resource sharing extremes: one big trusted stack -- separate untrusted stacks
 - Resource isolation impacts robustness -- leverage separation kernel guarantees
 - “Network Stack Virtualization” - one known approach to network data isolation
- Degrees of Assurance
 - Depending upon deployment environment, assurance requirements may vary
 - CC assurance levels, e.g., EAL 3, EAL 5, EAL 7
 - At higher levels, MILS PPs to include a formal description (see: 9th ICCG [3])

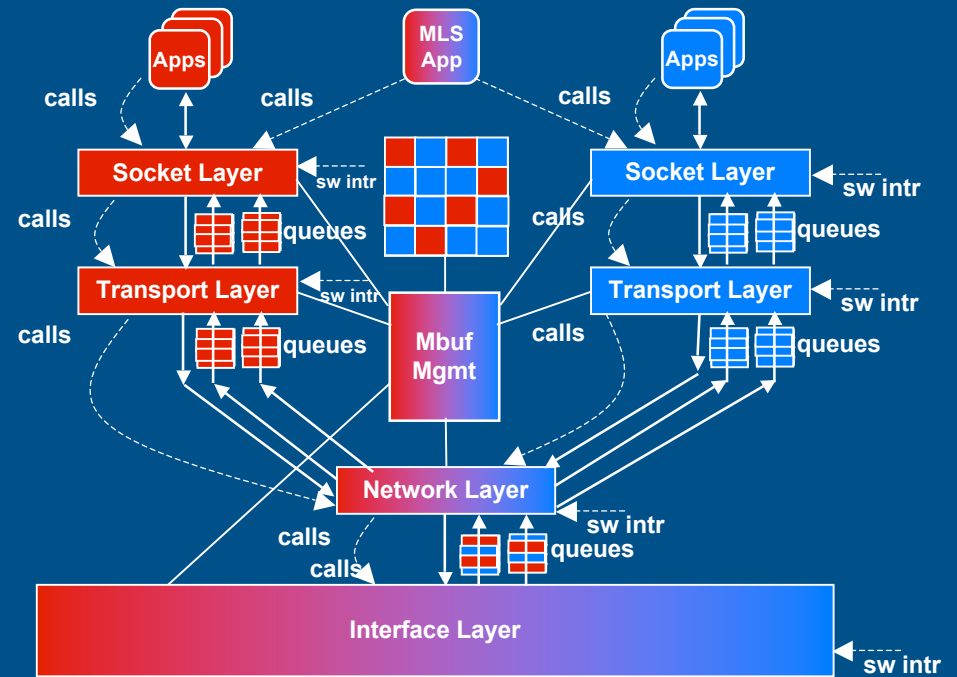
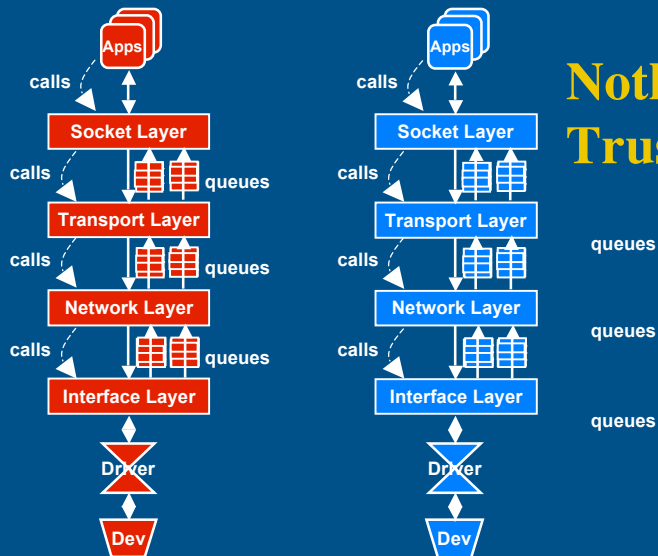
* CIPSO (Common IP Security Option - `92), CALIPSO (Common Architecture Label IPv6 Security Option - `09)

MILS Network System PP Challenges

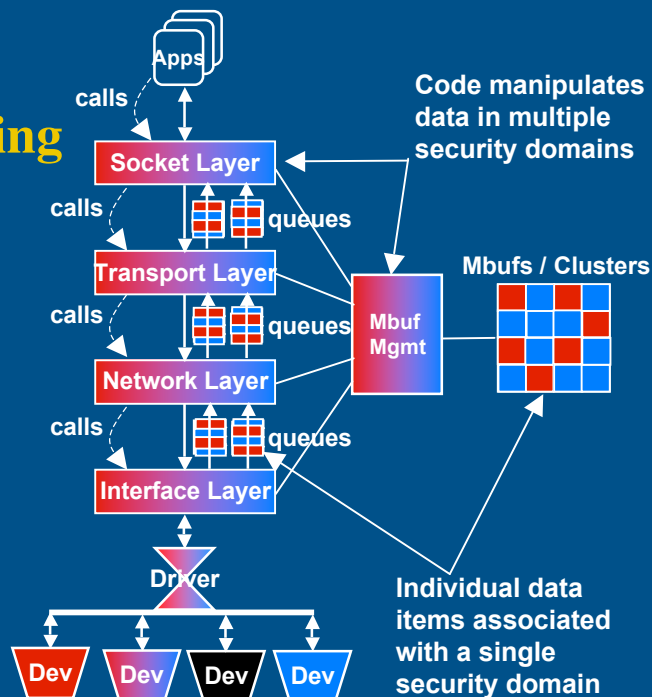
- Most complex product specified by a MILS PP so far (MILS Console System is a close second)
- For MILS conceptual integrity and interoperability, must avoid multiple, divergent protection profiles for MILS network product variations, e.g.,
 - MILS medium-assurance network system PP with many features
 - MILS high-assurance network system PP with limited features
- Desire to accommodate product families, product subsets
 - Multiple configurations within product family
 - Multiple assurance levels within product family
 - Multiple architectural approaches possible
 - Represented by a single MILS Network System PP (MNS PP)

Some architecture alternatives for MILS network system

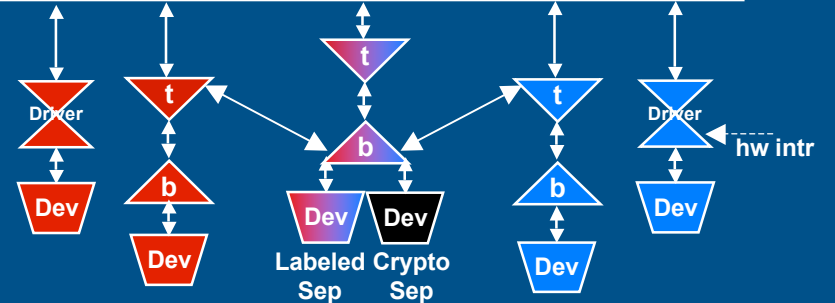
Nothing Trusted



Everything Trusted



Combination of Trusted and Untrusted



Novel Challenges for the MNS PP

- Initial attempts to represent as a single PP
 - Standard four CC component operations found to be inadequate
 - To **modularize** extensive functionality of IPv6
 - To avoid the **ugly alternative** of multiple PPs for function and assurance:

$mnpp^4_1$. . . $mnpp^4_n$ (EAL 4)

$mnpp^5_1$. . . $mnpp^5_m$ (EAL 5)

$mnpp^6_1$. . . $mnpp^6_p$ (EAL 6)

$mnpp^7_1$. . . $mnpp^7_q$ (EAL 7)

- **How many PPs** would we need to support product lines?
- **How many evaluations** need to be performed for a product line?
- Solution: MNS PP expressed by a “Polymorphic” PP
 - **Sub-Profiles**
 - **Sub-Profile Types**
 - Different Functionality / Assurance tradeoffs
 - Abstract formal model of the product presented **in more rigorous MILS PPs** (see: 9th ICC [3])

MNS “Sub-Profiles” and sub-profile types

- Sub-Profiles -- configuration building blocks
 - Functional subsets similar to draft DISA / Open Group standards for IPv6
 - Each assigned a sub-profile type
- Sub-profile Types (A, B, C) -- capture “complexity” considerations
 - Complexity of features: (low, moderate, high)
 - Real-time requirements: (hard, soft, none)
 - C --> A : Incrementally increasing assurance requirements
 - Increased scope, depth, and rigor
 - Added constraints on design and implementation structure



Type A

- Low feature complexity
- High assurance, formal
- E.g., EAL 7
- Hard real-time
- Virtualized stacks, SK
- E.g., robust devices

Type B

- Moderate feature complexity
- Medium-high assurance
- E.g., EAL 5
- Soft real-time
- Partially virtualized stacks
- E.g., infrastructure, military

Type C

- High feature complexity
- Medium assurance
- E.g., EAL 3
- No real-time requirement
- Conventional implementation
- E.g., server farm

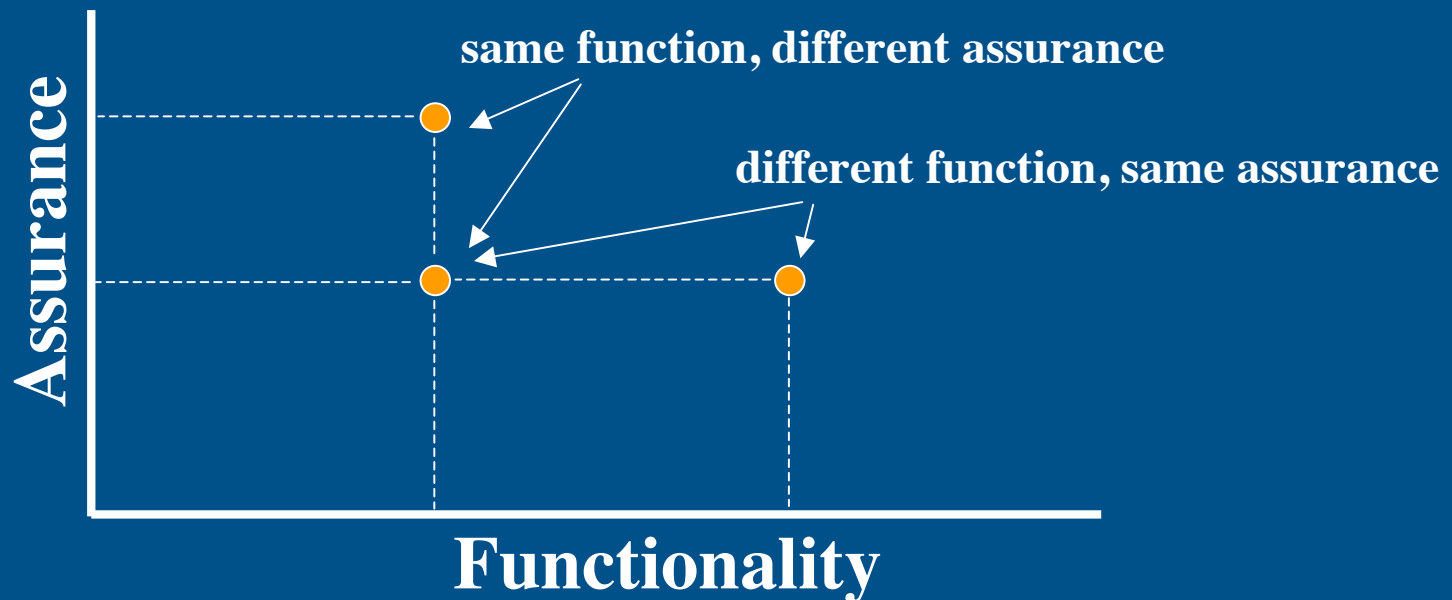
Sub-profile composition leads to product classes and subclasses.

Description of Polymorphic PPs

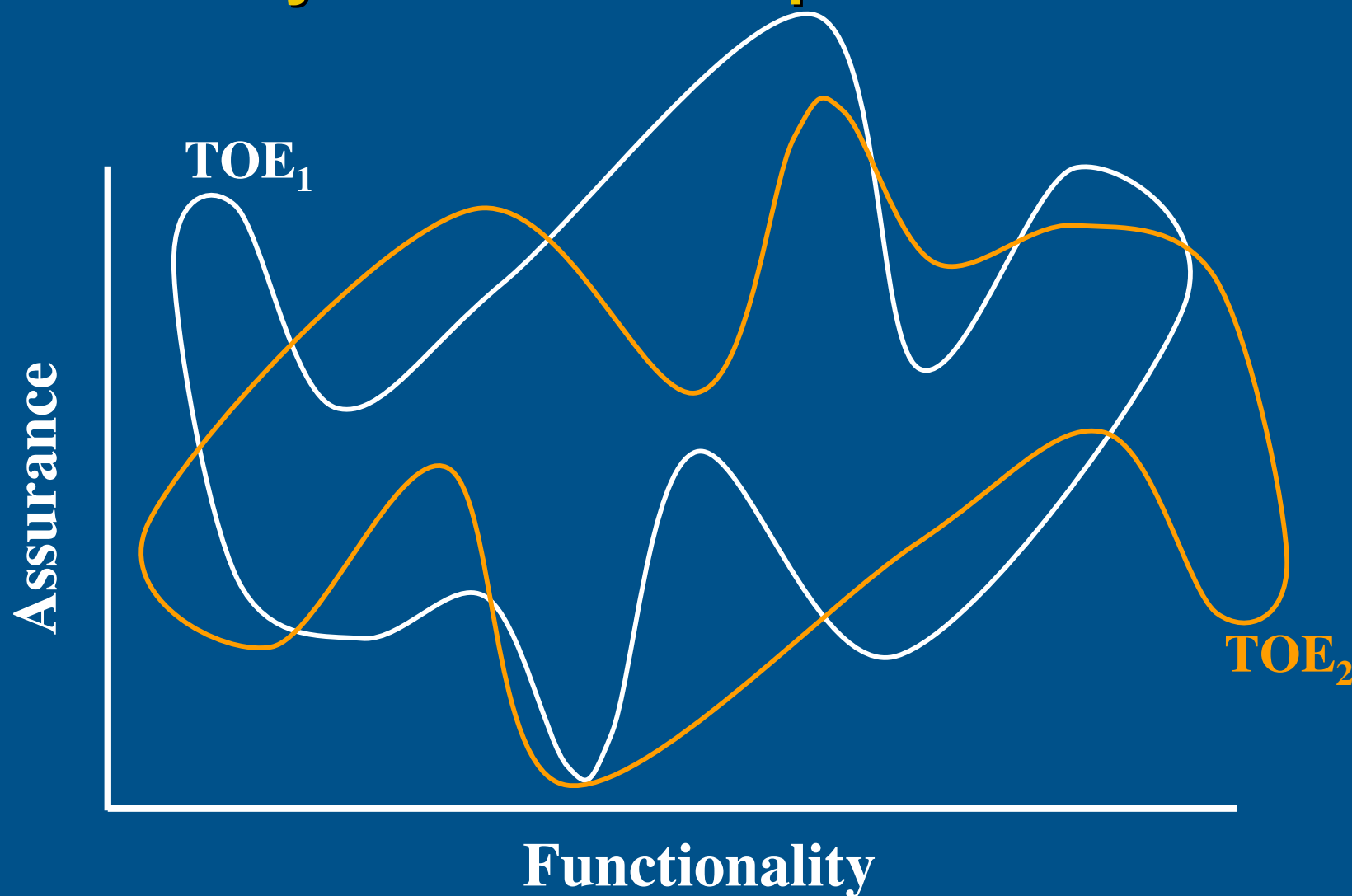
From Protection Profiles to Polymorphic Protection Profiles

A view of Common Criteria Protection Profiles

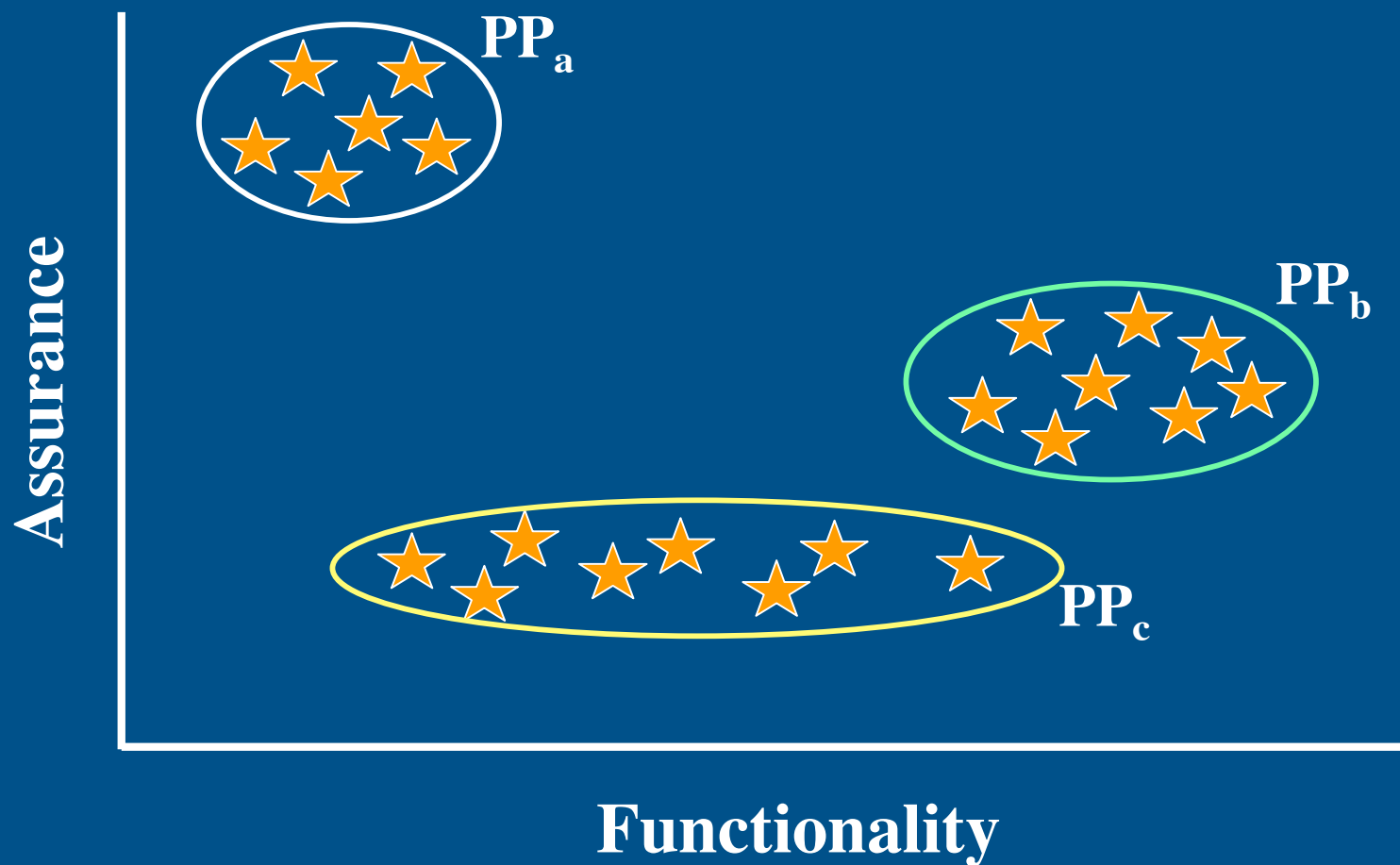
- A framework for the expression and justification of security functional and assurance requirements specifications
- A set of functions implemented with a degree of assurance
- Provide independent functional and assurance dimensions



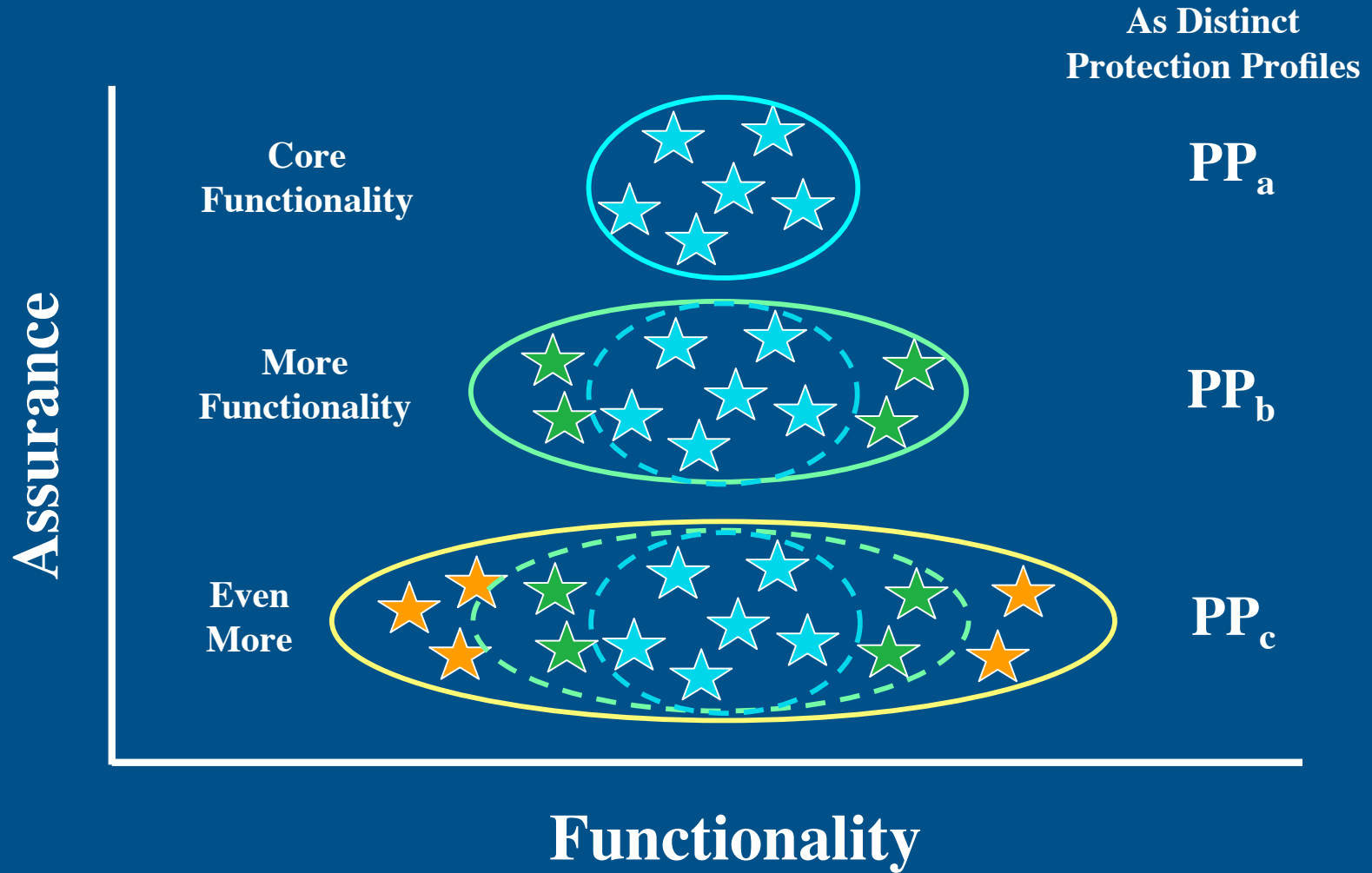
**Without protection profiles:
TOEs would be arbitrary sets of points in the
Functionality \times Assurance space**



Protection Profiles constrain the Functionality × Assurance space

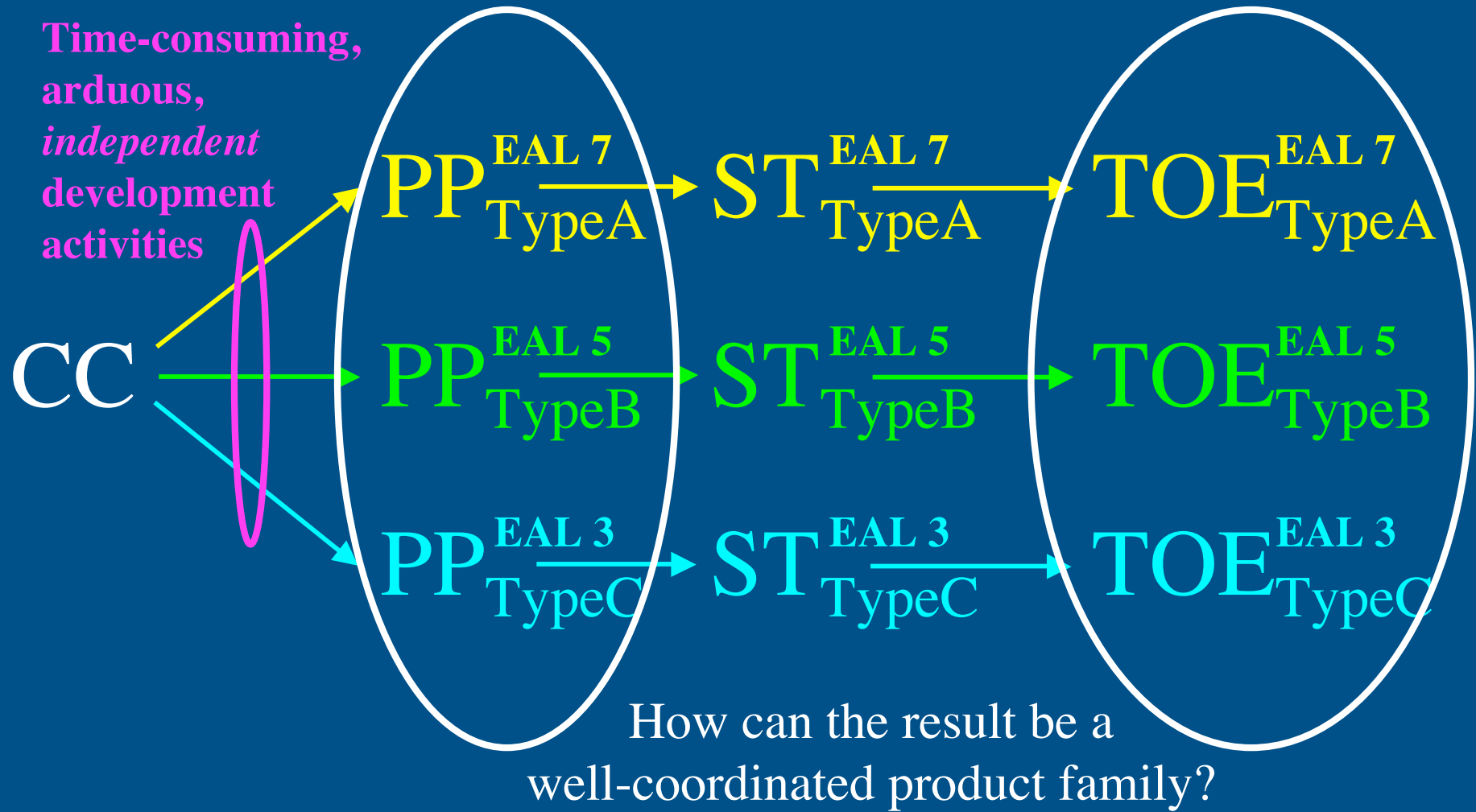


Functionality versus Assurance Tradeoff considers cost and tractability of TOEs

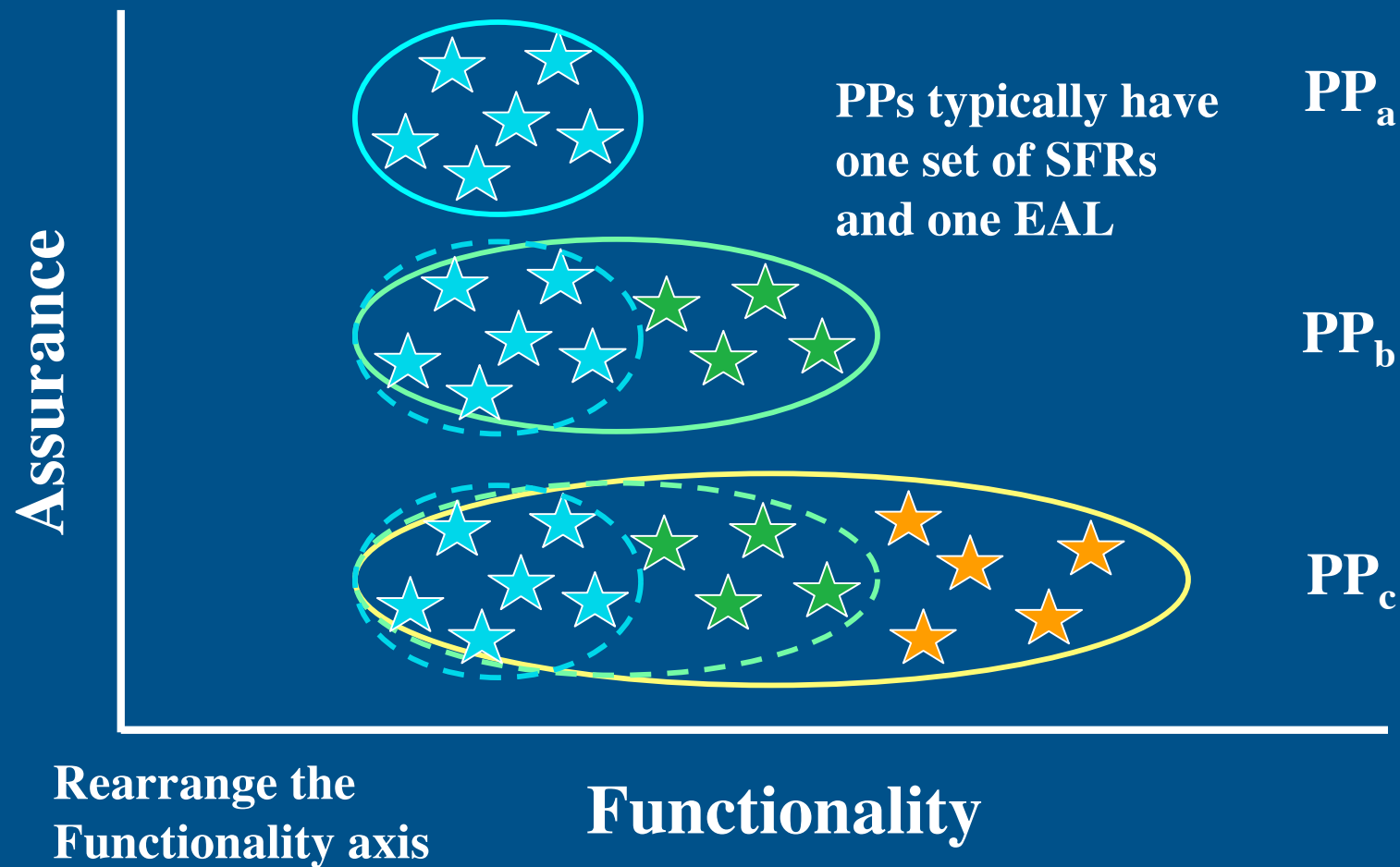


Risk associated with independent development of PPs for product family members

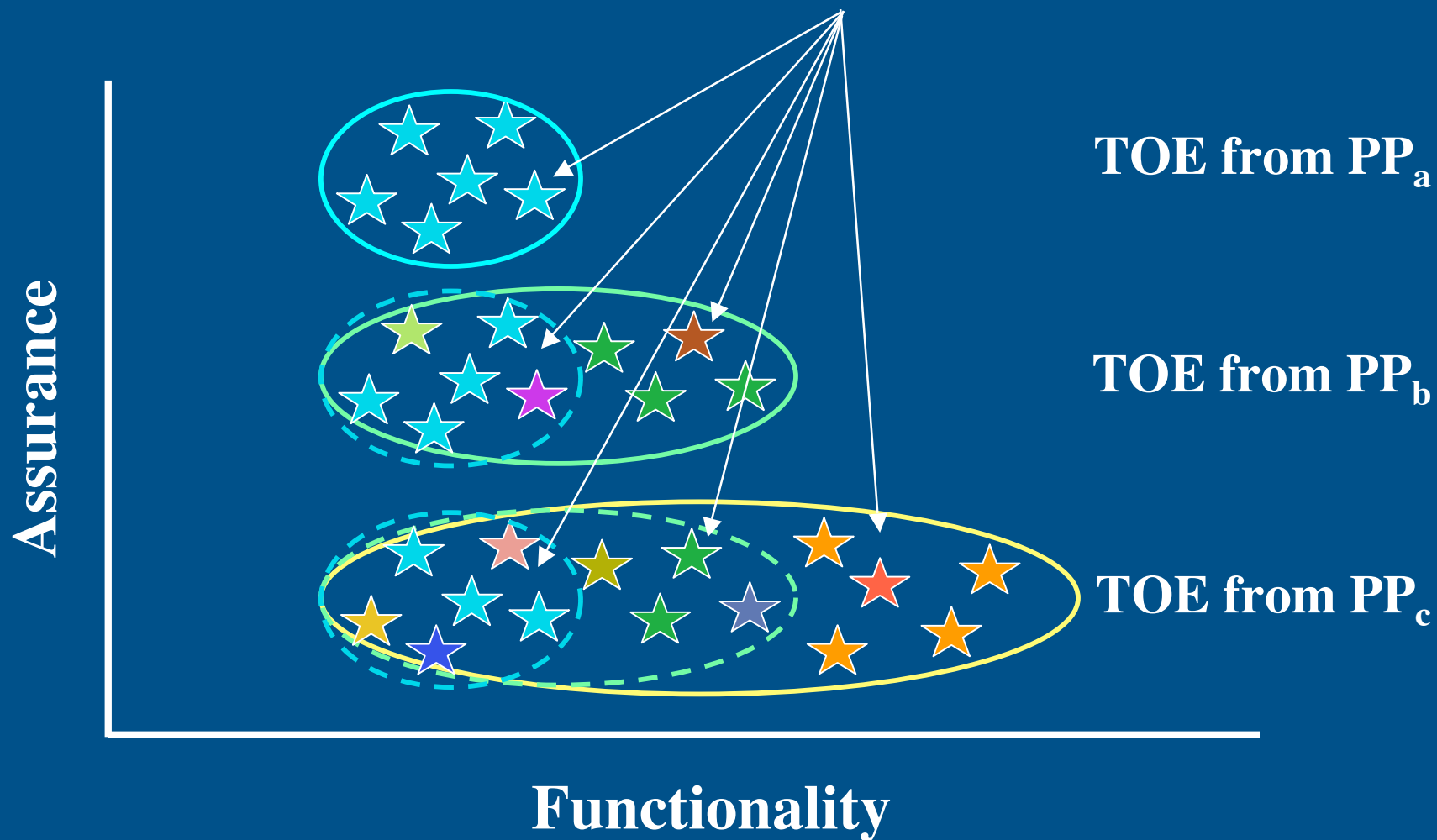
What's the likelihood these PPs will be coherent if independently developed?



Is there an alternative to Multiple Protection Profiles for Members of a Product Family?



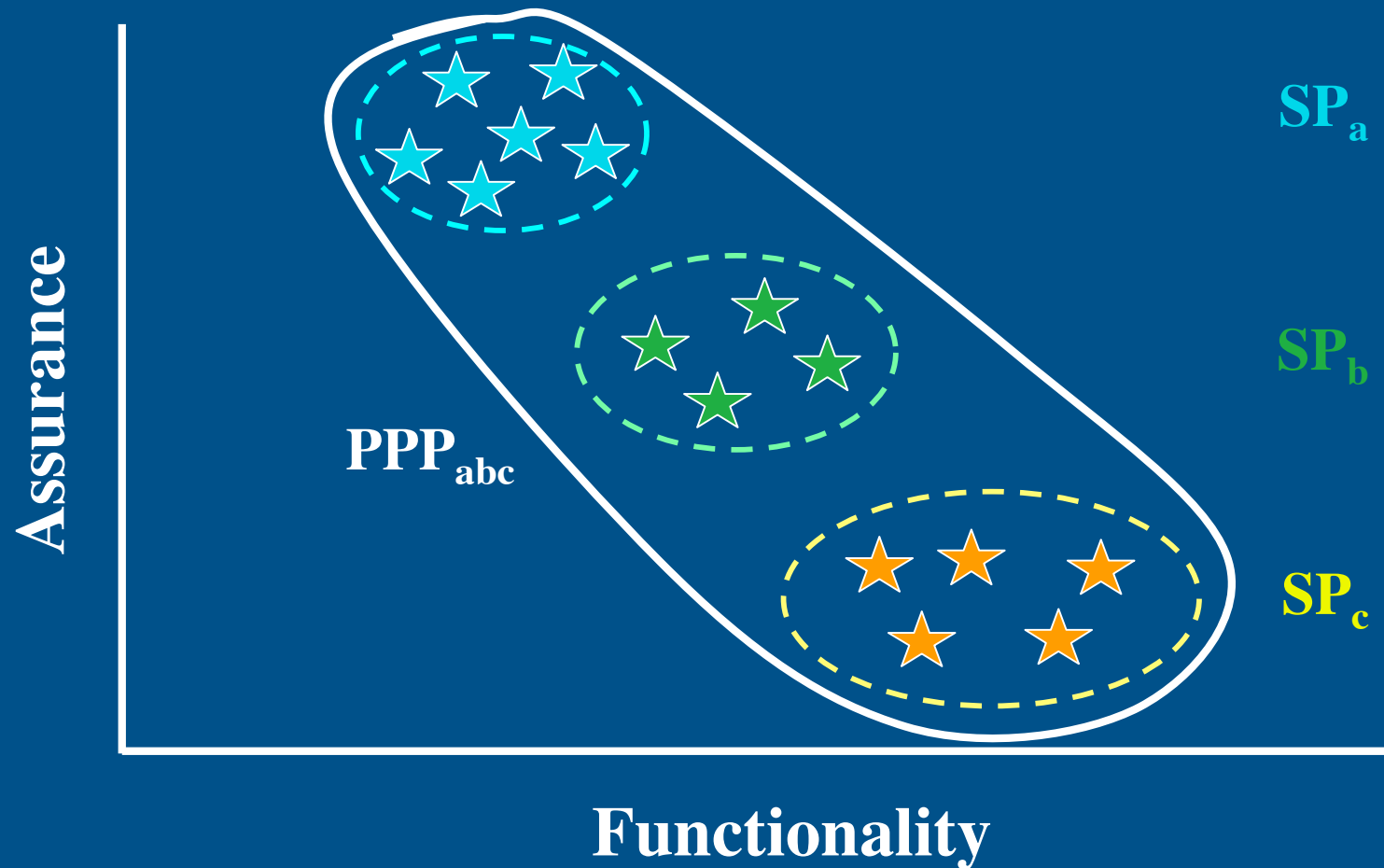
Independently developed PPs could lead to irreconcilably divergent products



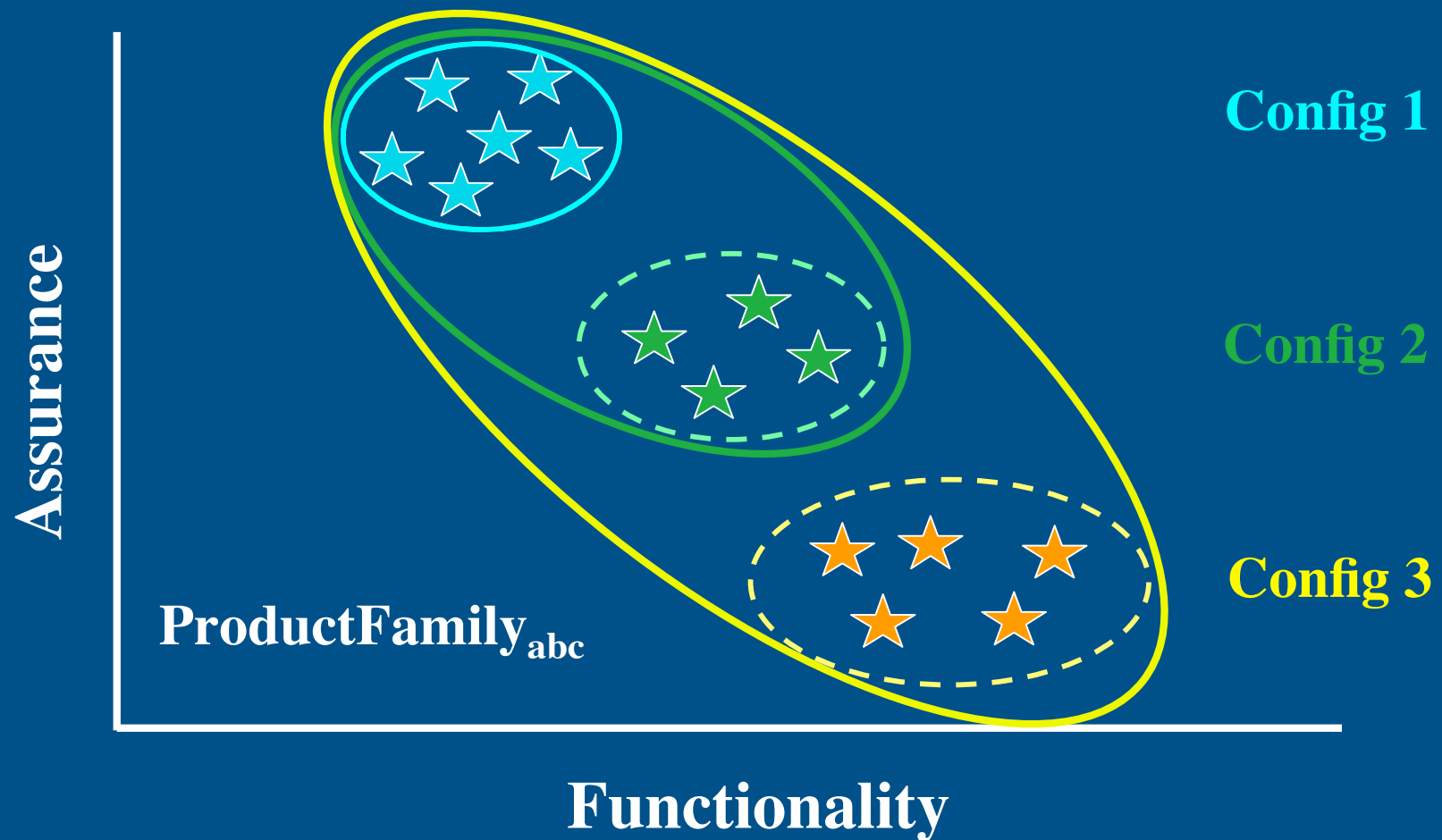
Objectives for Polymorphic PPs

- Keep all the security requirement information for product family in one PPP document
 - Vendors may build product families around a PPP
 - Permits greater flexibility in applying the PPP to derive TOEs
 - Greater range of functional combinations
 - Range of assurance options
 - Vendors can identify and target their “niche”
 - Vendors may compete on the basis of assurance as well as functionality
 - Support coherence of derived PP/ST/TOEs
- Commonality among members of a product family leveraged
 - By developers
 - By evaluators
 - By system architects and integrators
- Maintain necessary coordination within the MILS vendor community
- Ultimately, save effort and money in the realization of MILS products
- Enable and use automation to support PP development and evaluation

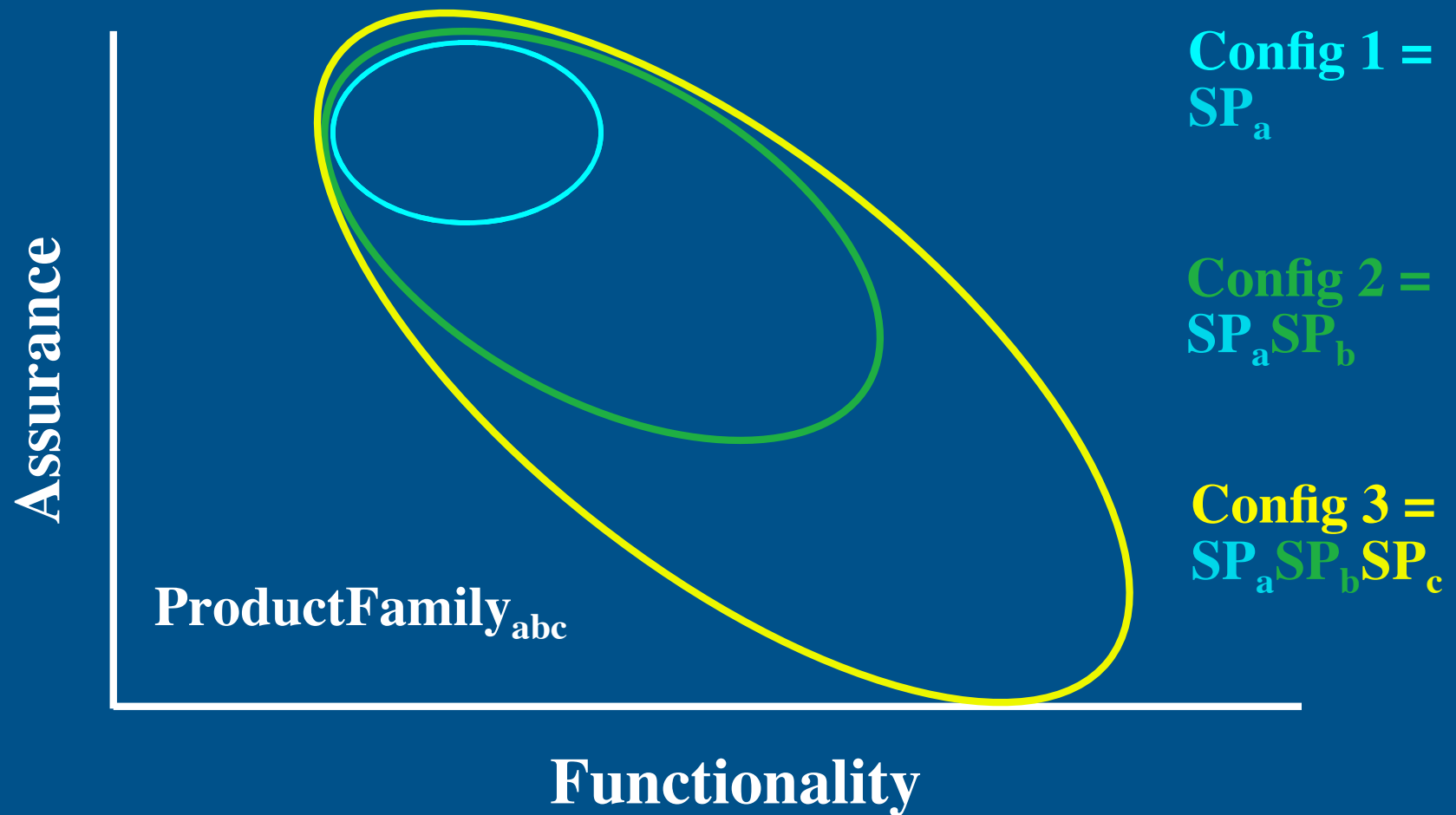
Polymorphic Protection Profile Approach: Factor Functionality × Assurance Clusters, Creating the “Sub-Profiles” of a PPP



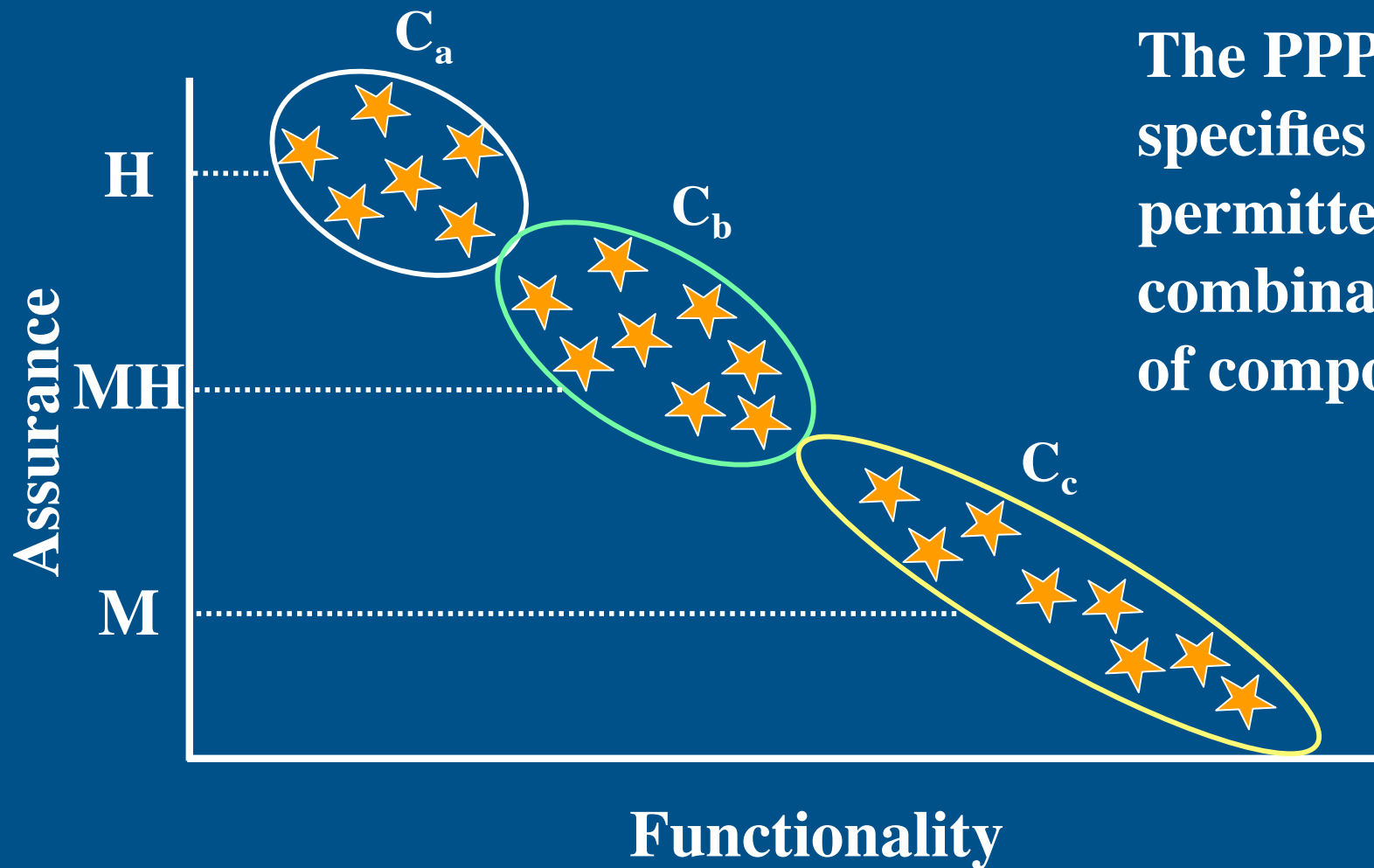
Then Define Different Configurations of a Product Family using the Sub-Profiles



Hierarchical Product Configurations



Components of a product family based on a single Polymorphic PP may be well-coordinated:



The PPP specifies the permitted combinations of components

Automation Support for PPPs

- Necessary to make Polymorphic PPs practical
 - Keep the security requirement information about a product family together
 - Automate the “projection” of PPs from a PPP
 - Perform automated consistency and completeness checks of PPP and “projected” PPs
 - To enable use of standard CC evaluation methodology
- Support from our Common Criteria Authoring Environment (CCAЕ) (see: 8th ICCС [2])
 - PPP a type of document known to the CCAЕ
 - Perform **consistency and completeness checks** on the PPP and PPs
 - Provide the **Projection Function** to generate PPs from the PPP
 - CCAЕ has an internal knowledge base and relational model of the CC
 - Knows about hierarchy and dependencies of components
 - Used by the PPP author to generate the PPs for PPP evaluation
 - And by evaluator to generate *differential work unit* checklists for efficient evaluation

Evaluating Polymorphic PPs

Evaluation of Polymorphic PPs (1)

- How can PPPs be evaluated?
 - We do not want to create a new and unfamiliar evaluation methodology
 - We want to leverage existing evaluation infrastructure, APE, and the CEM
- “Projection” of a polymorphic protection profile onto a set of standard PPs
 - The PPP *per se* is not directly evaluated by an evaluator
 - A **Projection Function** selectively projects-out a sub-profile combination
 - Use chosen sub-profile(s) and the other generic elements of the PPP
 - Create a standard protection profile with a single assurance level claim
 - Each sub-profile combination specified in the PPP projects a different PP
 - Conditions for **PPP well-formedness** to be assured by the PPP author:
 - The projection function is defined for all sub-profile combinations specified in PPP
 - All generated protection profiles are well-formed (complete and consistent)
- Managing the complexity of a PPP
 - PPP projection *could* be done manually - would be tedious and error prone!
 - The generated PPs will ordinarily have a high degree of commonality
 - Automation of the Projection Function
 - Generate **standard APE protection profiles** amenable to CEM
 - Generate **differential work unit checklists** for ordered PP evaluation

Evaluation of Polymorphic PPs (2)

This approach leverages established practice

The evaluation effort (to a first approximation) is linear in the # of instantiations of the PPP.

But the # of instantiations may be exponential in the number of sub-profiles.

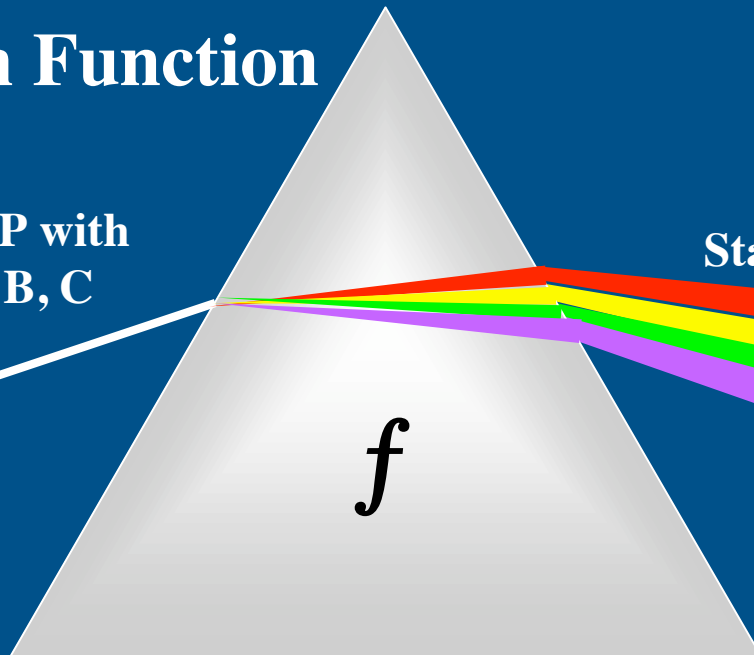
The PPP includes a list of explicitly permitted sub-profile configurations

Commonality among configurations reduces evaluation effort below linear

Projecting the PPP to standard PPs

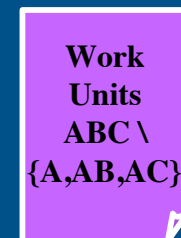
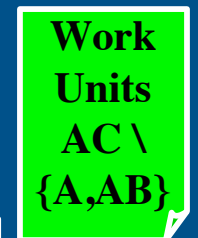
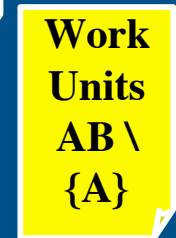
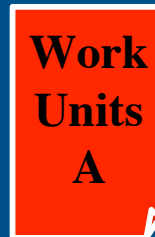
Projection Function

Polymorphic PP with sub-profiles A, B, C



Standard PPs

Evaluation Work Unit Checklists



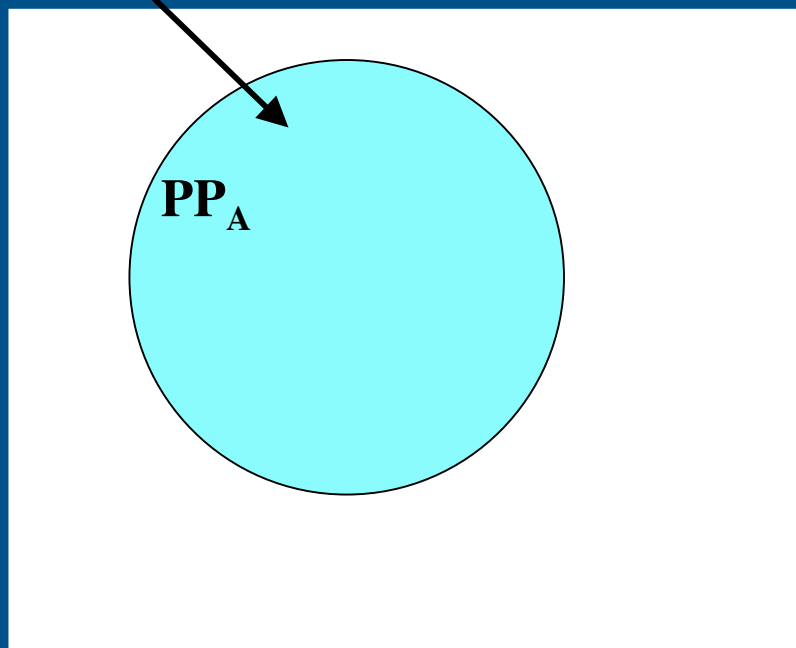
$$\begin{aligned}
 f \text{ PPP}_{ABC} \{ \{A\}, \{A,B\}, \{A,C\}, \{A,B,C\} \} \\
 = \{ \text{PP}_A, \text{PP}_{AB}, \text{PP}_{AC}, \text{PP}_{ABC} \} \\
 + \text{Evaluation Work Unit Checklists}
 \end{aligned}$$

Difference operator “\” applies comp’nt dependency, hierarchy, and other PP property closures. Differential work units assume ordered evaluation of PPs.

Evaluation differential work units (1)

Entailed work units to be performed to

evaluate $f \text{ PPP}_{ABC} \{A\} = \text{PP}_A$

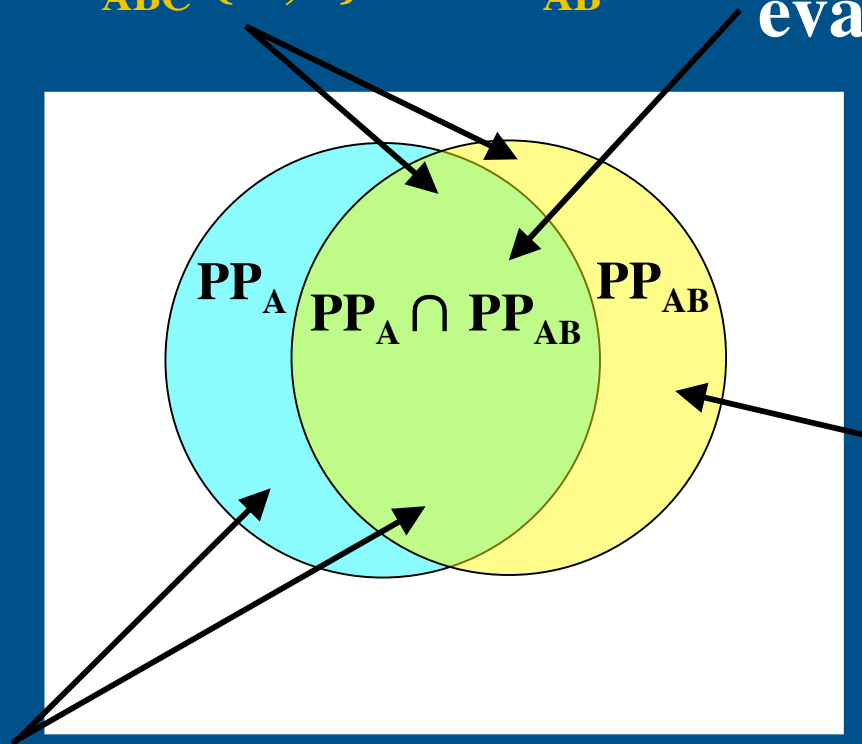


Note, the following Venn diagrams represent contents of projected PPs, not PPP sub-profiles.
Projected PPs may have substantial intersection, while sub-profiles may be disjoint.

Evaluation differential work units (2)

Work units entailed to evaluate f $PPP_{ABC} \{A,B\} = PP_{AB}$

PP_{AB} common work units completed for evaluation of PP_A

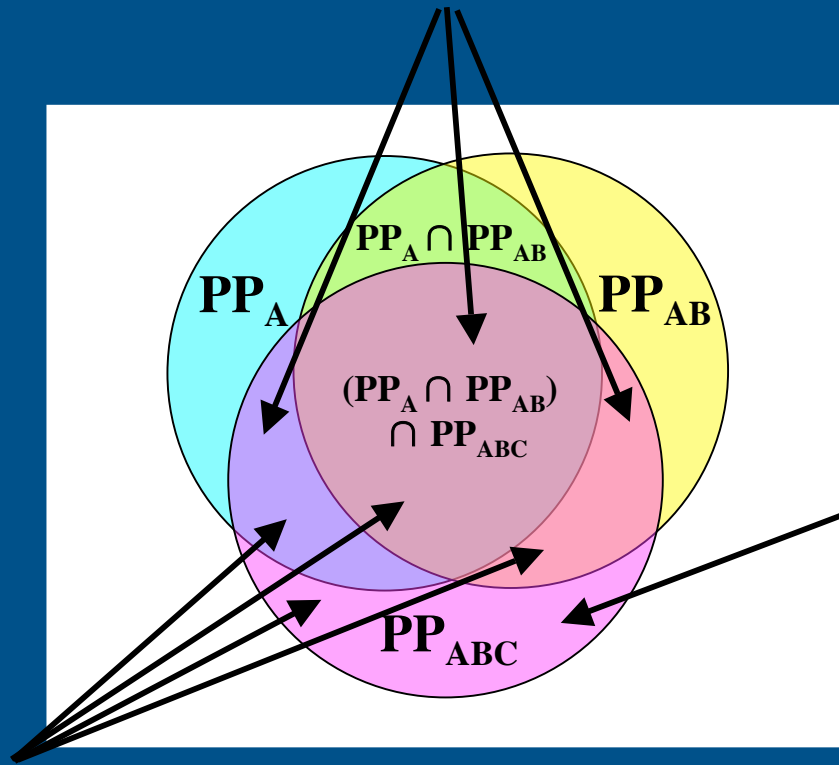


Differential work units $AB \setminus \{A\}$ to be performed to complete evaluation of PP_{AB}

Work units already completed during evaluation of PP_A

Evaluation differential work units (2)

PP_{ABC} common work units completed for evaluation of PP_A and PP_{AB}



Differential work units $ABC \setminus \{A, AB\}$ to be performed to complete evaluation of PP_{ABC}

Work units entailed to evaluate $f \text{ PPP}_{ABC} \{A, B, C\} = PP_{ABC}$

Using Polymorphic PPs

How are PPPs to be used?

- A developer may adopt a PP generated from the PPP, to develop a single conforming ST / TOE product
- A developer may create a polymorphic security target conforming to the PPP, generate conforming ST(s), and develop a TOE product family
- Members of a product family may have different functionality and assurance levels

Summary

- There is compelling business motivation for product families
- PP practice should and can support, rather than thwart, coherent product families
- PPPs can be defined in a way that yields the desired benefits without departing from the existing evaluation methodology
 - Sub-profiles collect function and assurance groupings
 - Multiple assurance levels as a result of sub-profile configurations
 - Explicit chosen configurations to limit evaluation effort
 - PPP is evaluated by evaluating the set of standard PPs generated
- Automation support makes PPPs practical
 - Generate and check “projected” PPs
 - Track and reuse cumulative work unit evaluation results

Acknowledgments

- The polymorphic protection profile concept arose from our work on MILS protection profiles, funded in part by the HAMES program (see below), and from our MILS research work at SRI International.
- John Rushby, Computer Science Laboratory, SRI International, principal investigator for the MILS research work.
- Carolyn Boettcher, Raytheon, manager of the HAMES (High-Assurance Middleware for Embedded Systems) project.
- Air Force Research Laboratory and the AF Cryptographic Modernization Program Office, sponsors of the HAMES project.
- LynuxWorks, for supporting this presentation at ICCC.

References

- [1] John Rushby and Rance DeLong. Compositional Security Evaluation: the MILS approach. In *8th International Common Criteria Conference*, Rome, Italy, 2007.
- [2] Rance DeLong and John Rushby. A Common Criteria Authoring Environment Supporting Composition. In *8th International Common Criteria Conference*, Rome, Italy, 2007.
- [3] Rance DeLong and John Rushby. High-Assurance Development and Evaluation: Rethinking the Common Criteria and EAL 7. In *9th International Common Criteria Conference*, Jeju, Korea, 2008.
- [4] Carolyn Boettcher, Rance DeLong, John Rushby, and Wilmar Sifre. The MILS Component Integration Approach to Secure Information Sharing. In *27th AIAA/IEEE Digital Avionics Systems Conference*, St. Paul, MN, Oct 2008, awarded *Best of Conference*.