

[소프트웨어] SSPL 플랫폼 세계표준 동향

대량맞춤생산(SSPL: Software and Systems Product Line)은 1760년 '유럽식 기술집약형 생산방식', 1870년 '미국식 기술분업형 생산방식', 1910년 '미국식 대량생산방식', 1980년 '유럽식 시장맞춤형 대량생산방식'의 과정을 거치면서 발전해 왔다. SSPL은, 제품과 서비스 생산에 필요한 전체 생명주기에 걸쳐서, 거버넌스, 사업가치, 프로세스, 플랫폼의 네 가지 역량과 '유럽식 시장맞춤형 대량생산방식'을 활용 하는 높은 성숙도의 SW와 시스템의 구현기술이다. 이 구현기술을 통해 SSPL은 구매자가 원하는 사양의 제품과 서비스를 저렴한 가격에 구입할 수 있게 해준다.

본고에서는 SSPL 플랫폼 역량 관련 세계 표준 동향에 중점을 두어 기술하고자 한다. 플랫폼이란 다른 기술이나 프로세스가 만들어지는 기반 기술로서 체계적으로 재사용되는 자산 체계를 의미하며, 구체적으로는 제품이나 서비스의 개발, 생산, 공통의 실행 환경을 포함 한다. 또한, 도메인 아키텍처, 설계 도구, 개발 도구, 형상 관리 도구, 테스트 도구, 구현 도구, 운영 도구와 이들 도구의 구축 및 활용 기술 등도 플랫폼 영역에 포함될 수 있다.

SSPL은 체계적으로 재사용 될 수 있는 내부 플랫폼을 기반으로 제품 라인을 개발함으로써 개발 비용과 개발 시간을 절약하면서 제품 라인을 통하여 다수 고객을 더 잘 만족시킬 수 있는 소프트웨어와 시스템의 융합 기술이다.

- 플랫폼화하면 다양한 고객이 필요로 하는 모든 특성(feature)을 개발자가 반복하여 개발하지 않아도 되므로 개발 비용과 개발 기간을 절약할 수 있고, 고객을 더 잘 만족 시킬 수 있는 제품의 기능과 품질 속성을 추가 개발할 수 있게 해준다.
- 또한, 플랫폼 기술은 다양한 제품으로부터 얻어진 기술과 지식이 공통의 플랫폼에 집적되어 지속적으로 발전하게 되므로 전체 제품군의 경쟁력을 향상시킬 수 있게 해준다.
- 반면에 제품 별 혹은 서비스 별로 개발되면 이러한 축적 효과가 약하고 제품수명이 짧아지는 경향이 강하게 된다.
- EU 등의 SSPL을 성공적으로 도입한 기업들은 플랫폼 구축에 많은 노력을 할애해왔다. 보쉬의 마켓 세그먼트 별로 구축한 3개의 플랫폼, 필립스에서 모든 영상장비에 공통으로 사용한 영상처리 플랫폼, 그리고 NH인더스트리스의 용이한 컴포넌트 조립을 지원하는 플랫폼 등이 그 사례이다. 이들 기업들은 공히 구축한 플랫폼에 맞추어 요구사항, 모델, 코드, 시험 등의 플랫폼 자산과 개별 제품들에서 사용할 자산들을 개발해왔다.
- SSPL은 제품 군 멤버들이 최대한 공유할 수 있는 플랫폼을 만들되 멤버들 간의 차별성을 보장할 수 있는 높은 수준의 플랫폼 개발 역량을 필요로 한다.

SSPL은 소프트웨어와 일반 시스템 개발에 모두 적용될 수 있으며, 또한 소프트웨어와 시스템이 융합된 소프트웨어 집약 시스템 (Software Intensive System)의 개발에도 잘 적용될 수 있다. 소프트웨어와 시스템의 융합은 점점 더 일반화 되어가고 있으며, 그 이유는 소프트웨어로 인하여 시스템에 지능이 부여됨으로써 시스템이 더 많은 문제를 더 잘 해결할 수 있게 되고, 이는 고객을 더 잘 만족시킬 수 있기 때문이다.

- 소프트웨어 집약 시스템이란 소프트웨어가 중요한 역할을 하는 시스템으로서, 소프트웨어 제품, 소프트웨어 서비스, 소프트웨어 기반 시스템 등을 포함한다. 따라서 제조분야, 국방분야, 통신분야 등의 임베디드 시스템도 소프트웨어 집약 시스템이라 볼 수 있다.

SSPL에서는 소프트웨어 부분의 개발에만 초점을 두지 않고 소프트웨어 집약 시스템 전체의 개발에 초점을 두고, 소프트웨어와 시스템을 통일된 관점에서 통합하여 개발함으로써 복잡도가 점점 높아지고 있는 시스템 개발을 더 체계적으로 지원한다. 또한, 소프트웨어 집약 시스템의 소프트웨어 특징으로 인하여 대량맞춤생산을 보다 풍부하게 지원할 수 있다.

2017년 5월 현재 우리나라의 주도로 준비되고 있는 SSPL 세계표준현황은 다음과 같다:

o 시스템 및 소프트웨어 융합 플랫폼을 위한 아키텍처 텍스처 (Architectural texture in the product line architecture):

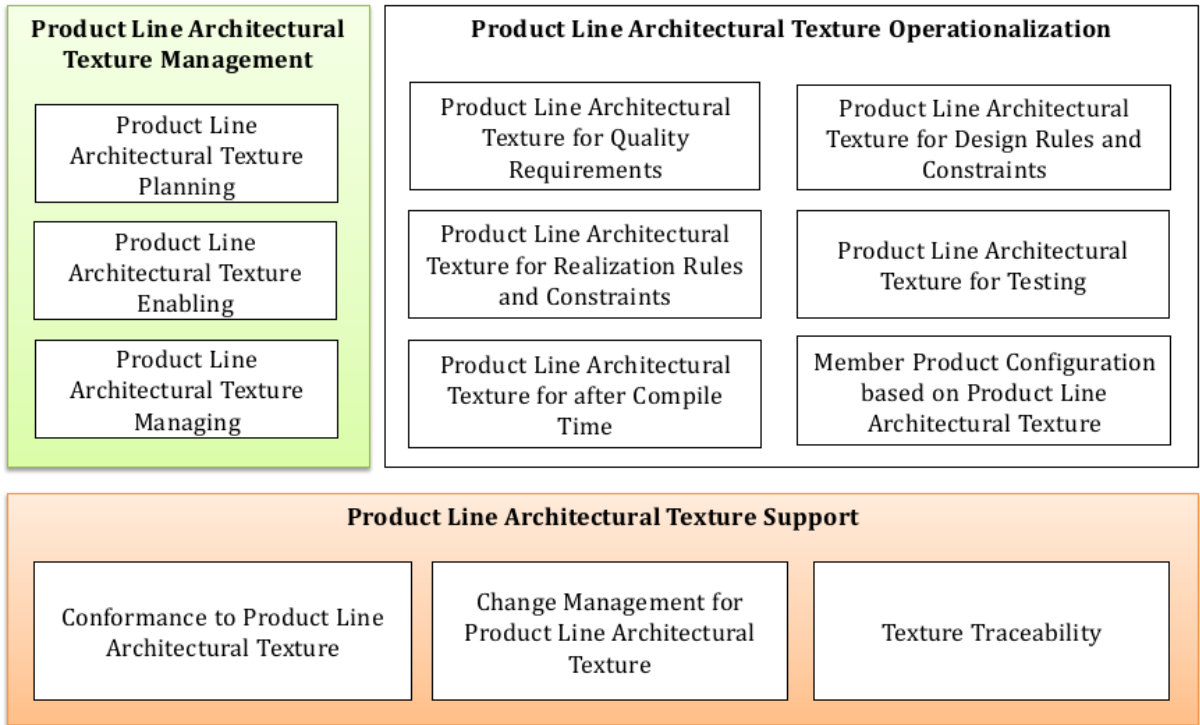
- 공통성과 가변성의 구조, 아키텍처단계와 구현단계 및 런타임에서 반드시 지켜야 할 가이드, 규칙 및 제약사항을 다룬다.

- 아키텍처 텍스처는 구현 가이드, 어스펙트(Aspect)와 컨선(Concern) 분리 규칙, 아키텍처 스타일과 패턴 사용, 컴포넌트가 제공(Provide)하는 인터페이스(provided interface)와 요구(Request)하는 인터페이스(required interface) 제약(Constraints), 코딩 컨벤션 등을 포함한다.

- SSPL에서 아키텍처 텍스처는 아키텍처가 가변성을 어떻게 다루며, 아키텍처를 사용하는 멤버제품들은 가변성을 어떻게 구현하고, 수정해야 하는지에 대한 규칙과 제약사항을 제공해야 한다.

- [그림 1]의 핵심 프로세스가 보여주는 바와 같이 SSPL에서 아키텍처 텍스처는 품질 요구사항을 시작으로, 가변성이 가장 많이 정의되는 설계 단계에서 정의된다. 텍스처는 아키텍처 설계 단계 이외에도 구현, 컴파일 이후에 가변성을 실제 구현하고 실행 가능한 산출물을 생성하는데 준수해야 할 모든 사항을 상세하게 기술해야 한다. 왜냐하면 가이드, 규칙 및 제약조건을 준수하여 구현하고 수정하지 않으면 여러 멤버제품들이 아키텍처를 공유하기 어렵기 때문이다.

- SSPL에서 아키텍처 텍스처는 가변성 관련 규칙과 제약을 핵심으로 다루기 때문에 가변성 모델, 가변성 관련 산출물과의 추적성이 세심하게 관리되고 유지되어야 한다.



[그림 1] SSPL 아키텍처 텍스처 핵심 프로세스

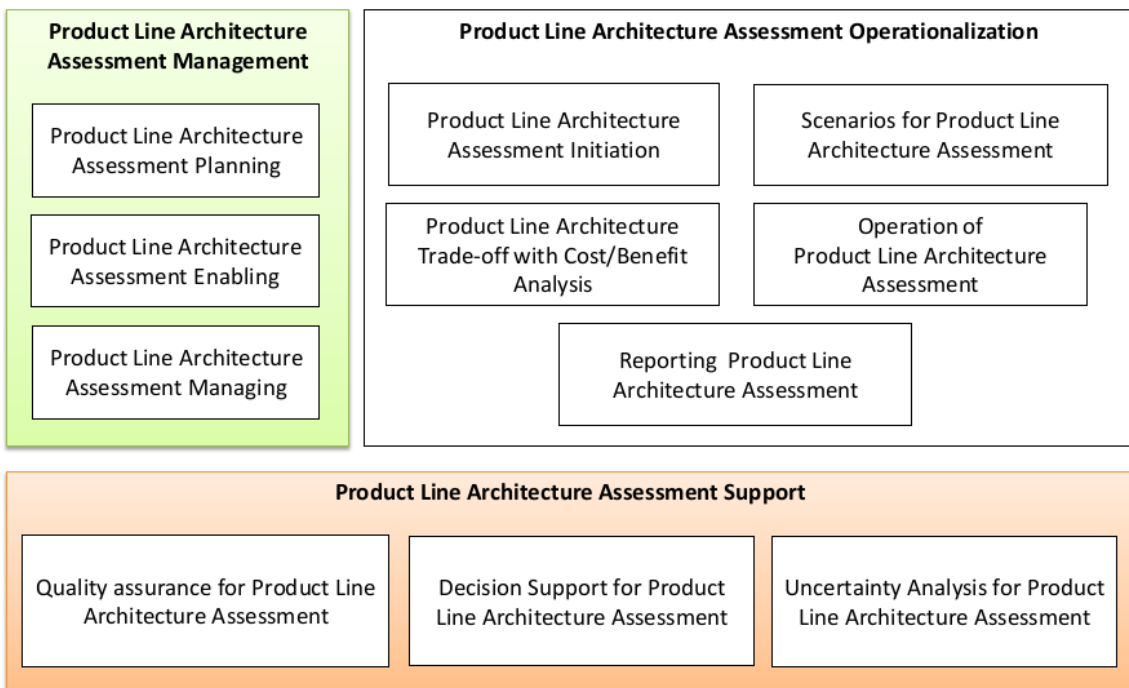
Process	Subprocess	Task
Product line architectural texture management	Product line Architectural texture planning	<ul style="list-style-type: none"> ● Identify architectural texture strategies ● Assign responsibility for architectural texture operationalization ● Define success measures for architectural texture operationalization ● Estimate adequate resources needed for architectural texture operationalization ● Document architectural texture plans
	Product line architectural texture enabling	<ul style="list-style-type: none"> ● Enable architectural texture environment ● Provide guidance for architectural texture operationalization ● Enable measurement environment for quantifying architectural texture operationalization
	Product line architectural texture managing	<ul style="list-style-type: none"> ● Review architectural texture operationalization status with success measures ● Control issues on architectural texture operationalization ● Make corrective actions on architectural texture operationalization

Product line architectural texture operationalization	Product line architectural texture for requirements	<ul style="list-style-type: none"> ● Identify architecturally significant requirements supported by the texture ● Analyze common rules and constraints to deal with variability in requirements ● Define architecture drivers for the texture
	Product line architectural texture for design rules and constraints	<ul style="list-style-type: none"> ● Analyze common rules and constraints for consistent architecture design and reuse ● Formulate the texture for architecture design ● Verify the texture for architecture design
	Product line architectural texture for realization rules and constraints	<ul style="list-style-type: none"> ● Analyze common rules and constraints for guiding realization ● Formulate the texture for realization ● Verify the texture for realization
	Product line architectural texture for testing	<ul style="list-style-type: none"> ● Analyze common rules and constraints for guiding testing ● Formulate the texture for testing ● Verify the texture for testing
	Product line architectural texture for after compile time	<ul style="list-style-type: none"> ● Analyze common rules and constraints for guiding after compile time ● Formulate the texture for after compile time ● Verify the texture for after compile time
	Member product configuration based on product line architectural texture	<ul style="list-style-type: none"> ● Analyze product line architectural texture for member product configuration ● Formulate member product configuration under the adherence of the texture ● Verify the texture for member product configuration
Product line architectural texture support	Conformance to product line architectural texture	<ul style="list-style-type: none"> ● Objectively assure the conformance to the texture in relevant domain engineering process ● Objectively assure the conformance to the texture in relevant application engineering process ● Communicate and resolve nonconformity issues ● Establish records on conformity assurance activities
	Change management for product line architectural texture	<ul style="list-style-type: none"> ● Identify changes on product line architectural texture ● Analyze change impacts on relevant domain engineering artefacts ● Analyze change impacts on relevant application engineering artefacts ● Record and communicate the change status of the texture

	Texture traceability	<ul style="list-style-type: none"> ● Define trace links between texture and its relevant artifacts ● Define trace links between texture and its relevant variability elements in variability model ● Identify and resolve issues on texture traceability ● Record and communicate the traceability status of the texture
--	----------------------	--

○ 시스템 및 소프트웨어 융합 플랫폼을 위한 아키텍처 평가(Product line architecture assessment):

- 제품라인 아키텍처가 제품라인 관련 품질 요구사항을 충족시킬 수 있는지 확인하는 품질 보증 기술로, 제품라인 아키텍처가 오랫동안 유효하기 위해서는 품질 요구사항을 잘 충족시키도록 정의되어야 한다.
- SSPL 아키텍처 평가는 플랫폼 아키텍처 구조와 텍스처가 공통성을 잘 지원할 수 있고, 정의된 가변성을 유연한 구조로 잘 반영하고 있으며, 필요한 품질 요구사항을 충족시킬 수 있는지를 평가한다. 특히, 여러 멤버제품이 공유하기 때문에 텍스처가 필요한 수준으로(just enough) 명확하게 정의되었는지 확인하고 보증할 수 있어야 한다.
- [그림 2]의 핵심 프로세스가 보여주듯이 SSPL 아키텍처 평가는 단일 제품과 달리 가변성을 포함한 제품라인 품질 요구사항, 공통성/가변성을 포함하고 있는 제품라인 아키텍처를 대상으로 하기 때문에 평가를 위한 시나리오, 불확실성 분석, 비용/이익 평가, 위험 분석 등에서 단일제품과는 차별화된 방법과 도구를 필요로 한다.



[그림 2] SSPL 아키텍처 평가 핵심 프로세스

Process	Subprocess	Task
Product Line Architecture Assessment Management	Product line architecture assessment planning	<ul style="list-style-type: none"> ● Identify PL architecture assessment strategies ● Assign responsibility for PL architecture assessment operationalization ● Define success measures for PL architecture assessment operationalization ● Estimate adequate resources needed for PL architecture assessment operationalization ● Document PL architecture assessment plans
	Product line architecture assessment enabling	<ul style="list-style-type: none"> ● Enable PL architecture assessment environment ● Provide guidance for PL architecture assessment operationalization ● Enable measurement environment for quantifying PL architecture assessment operationalization
	Product line architecture assessment managing	<ul style="list-style-type: none"> ● Review PL architecture assessment operationalization status with success measures ● Control issues on PL architecture assessment operationalization ● Make corrective actions on PL architecture assessment operationalization
Product Line Architecture Assessment Operationalization	Product line architecture assessment initiation	<ul style="list-style-type: none"> ● Identify PL architecture assessment participants ● Mobilize resources for PL architecture assessment operation ● Initiate PL architecture assessment operation ● Perform preliminary PL architecture assessment
	Scenarios for product line architecture assessment	<ul style="list-style-type: none"> ● Develop scenarios for dealing with commonality in quality requirements ● Enhance scenarios for dealing with variability in quality requirements ● Prioritize scenarios for product-line-related quality requirements
	Product line architecture trade-off with cost/benefit analysis	<ul style="list-style-type: none"> ● Analyze architectural approaches on product-line-related quality requirements for trade-off analysis ● Analyze tradeoff with cost/benefit among architectural approaches for commonality in quality requirements ● Analyze tradeoff with cost/benefit among architectural approaches for variability in quality requirements ● Document the result and rationale of PL architecture trade-off with cost/benefit analysis

	Operation of product line architecture assessment	<ul style="list-style-type: none"> ● Perform aspect-oriented PL architecture assessment ● Perform PL architecture assessment ● Perform integrated PL architecture assessment ● Verify the alternative results of PL architecture assessment ● Provide the level of business goal satisfaction for high-priority alternatives ● Recommend selected architectural approaches
	Reporting product line architecture assessment	<ul style="list-style-type: none"> ● Collate product line architecture assessment results ● Document selected architectural approaches for commonality in quality requirements ● Document selected architectural approaches for variability in quality requirements ● Report and communicate PL architecture assessment results
Product Line Architecture Assessment Support	Quality assurance for product line architecture assessment	<ul style="list-style-type: none"> ● Objectively evaluate PL architecture assessment activities ● Objectively evaluate PL architecture assessment work products ● Communicate and resolve noncompliance issues ● Establish records of quality assurance activities for PL architecture assessment
	Decision support for product line architecture assessment	<ul style="list-style-type: none"> ● Establish full of references to PL architecture assessment ● Verify PL architecture assessment scenarios ● Validate PL architecture assessment decisions
	Uncertainty analysis for product line architecture assessment	<ul style="list-style-type: none"> ● Identify potential uncertainties on architecture assessment ● Analyze uncertainties in architectural approaches for commonality in quality requirements ● Analyze uncertainties in architectural approaches for variability in quality requirements ● Reflect uncertainties to relevant scenarios and approaches

이단형 (한국소프트웨어기술진흥협회 회장. danlee@kaist.ac.kr)