# Symverse Contract Template SCT - 내장 계약 함수



## **INDEX**

### 1. 개요

개요 용어 사전

#### 2. SCT 소개

SCT 계층 구조 SCT 사용자 구조 주요 기능 Table 쿠폰 서비스 예시

### 3. SCT 처리 방식

SCT 처리 프로세스 SCT 자료 구조



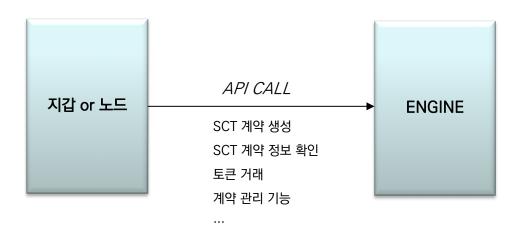
# 개요

- 1) 개요
- 2) 용어 사전



#### ❖ 개요

- 1. 스마트 컨트랙트 중 현재 가장 많이 사용되고 있는 항목을 내장화 하여 간편하게 생성 및 이용 가능
- 2. 개발 환경의 개선
  - 스마트 컨트렉트 전용 언어를 사용하지 않음 개발자 수급의 용이
  - 단순 API 호출로 개발 가능 개발 시간 단축
- 3. 유지 보수 비용 감소
  - gsym 업그레이드만으로 기능 개선 가능



추가 개발 없이 제공되는 API로 단순 처리 가능



#### ❖ 용어 사전

• SCT (SymVerse Contract Template)

Engine에 내장되어 있는 처리되는 계약 템플릿을 의미 ( 아래에 나오는 Token, Coupon 모두 SCT 기반 계약 )

• SCT20

SCT로 생성한 계약으로 balance의 가치가 동등 해 양으로 표현 가능한 토큰 (ERC20 Token like)

SCT30

SCT로 생성한 계약으로 balance 가 각각의 고유 가치를 나타낼 수 있는 토큰 (ERC721 Token like)

5

• SCT40 (Coupon)

SCT로 생성한 계약으로 쿠폰의 관련된 계약을 처리할 수 있는 쿠폰 형태의 계약

Coupon Item

SCT30,40 에서 생성된 고유 가치를 가지고 있는 쿠폰 정보

CreateSymId

SCT로 생성한 계약의 생성 Symld를 의미, 최고 권한을 가지고 있음

OwnerSymId

SCT로 생성한 계약의 소유 Symld를 의미, 계약의 소유권을 주장 할 수 있음



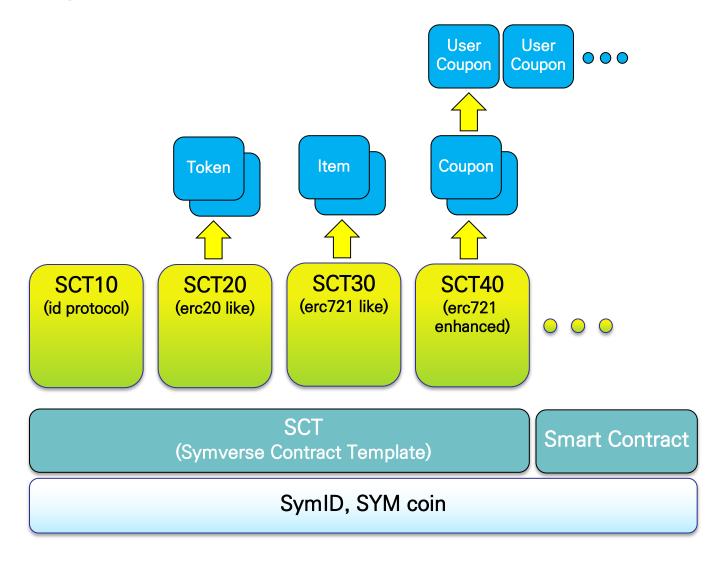
## SCT 소개

- 1) SCT 계층 구조
- 2) SCT 특징
- 3) 주요 기능 TABLE
- 4) 서비스 예시



## ❖ SCT(Symverse Contract Template) 계층 구조

추상화된 SCT 계층 구조입니다.





#### ❖ SCT 특징

Solidity와 SCT의 계약 생성 방식에 따른 비교

## Solidity를 통한 계약 생성 Solidity 코딩 Solidity 컴파일 계약 Transaction 발행 dApp 개발 시 Solidity에 대한 지식 필요 • Solidity에 대한 이해도가 낮을 경우 보안 이슈 발생 • Solidity 코딩 및 컴파일 등 여러 단계를 거치기 때문에 실제 서비스 개발 시 다양한 도구를 사용해야 함 dApp 개발의 속도 저하 및 어려움 / dApp 생태계의 성장 속도 저하

#### SCT를 통한 계약 생성

#### SCT 계약 생성 API 호출



#### 계약 Transaction 발행

• 별도의 선행 학습 필요 없이 dApp 개발 가능

8

- 단순 생성만으로도 engine에서 내부 보안 코드 적용
- 단순 API 호출만으로 dApp 서비스를 구축할 수 있음

쉬운 dApp 개발 / dApp 생태계의 성장 속도 상승



## ❖ SCT 주요 기능 TABLE

Creator : 계약 생성자 Owner: 계약 소유자 User: 일반 사용자

분류		기능	권한			설명	비고
			Creator	Owner	User		_ <u>-</u>
Common		SCT 생성		_		SCT 계약을 생성합니다.	
		계약 사용 제어 & 허가	0	0		SCT 계약의 상태를 제어합니다.	
Commi	OH	계약 소유자 이전	0			생성된 계약의 소유자를 이전합니다.	
		계약 정보 확인	0	0	0	SCT 계약의 정보를 확인합니다.	
	20	토큰 전송			0	특정 사용자에게 토큰을 전송합니다	
		토큰 제 3자 위임			0	특정 사용자에게 본인의 토큰 일부를 위임합니다.	
		토큰 제 3자 전송			0	위임 받은 토큰을 특정 사용자에게 전송합니다.	
		토큰 추가 발행	0	0		토큰의 양을 추가 발행합니다.	
		토큰 태움	0	0		토큰의 양을 일부 태웁니다.	
	30	아이템 대량 생성	0			쿠폰 아이템의 상태 확인	
SCT		아이템 확인	0	0		아이템의 정보를 확인합니다.	
301		아이템 전송	0	0	0	아이템을 특정 사용자에게 전송합니다.	
		보유 아이템 확인	0	0	0	특정 사용자의 보유 아이템을 확인합니다.	
	40	쿠폰 아이템 대량 생성	0			쿠폰 아이템을 대량 생성	
		쿠폰 아이템 확인	0	0	0	쿠폰 아이템의 상태 확인	
		쿠폰 아이템 상태 변경	0	0		쿠폰 아이템의 상태를 변경합니다.	예) 쿠폰 사용 처리 / 쿠폰 잠금
		쿠폰 아이템 전송	0	0	0	쿠폰 아이템을 특정 사용자 소유로 이전	
		보유 쿠폰 확인	0	0	0	특정 사용자의 보유 쿠폰을 확인합니다.	

9



#### ❖ SCT 서비스 예시

아래 그림은 Symberse dApp 서비스 모델 중 하나입니다.



dApp 서비스 사용

지갑 결제 요청



SCT 계약 생성 요청

SCT 계약 제공



Creator

- 일반적인 지갑을 통해 dApp을 이용하고 거래를 시도하는 사용하는 유저
- 유저는 클라이언트를 통해 원하는 서비스를 제공 받습니다.
- 유저 또한 SCT를 통해 단순 계약을 발행할 수 있으며 그에 따른 권한 또한 모두 유저가 갖고 있습니다.

- 클라이언트는 유저에게 서비스를 제공 하는 역할을 합니다.
- 클라이언트는 서비스 제공자를 통해 유저에게 서비스할 SCT 계약을 요청 합니다.
- 여기서 클라이언트는 계약의 소유자 즉, OwnerSymld 를 갖게 됩니다.
- 실제 데이터(주문, 상품, 쿠폰)등은 클라이언트에서 관리하고 그에 따른 SCT 상태는 Symverse 엔진을 통해 검증 및 유저에게 제공합니다.

10

- 서비스 제공자는 다양한 계약 생성 서비스를 클라이언트 들에게 제공하며 SCT 서비스 도구의 역할을 합니다.
- 서비스 제공자는 신뢰할수 있는 기관이며 서비스 제공자를 통해 생성된 모든 SCT 계약에 대한 통제권을 갖고 있습니다.
- 클라이언트가 SCT계약 요청에 대한 생성을 하게 되며 서비스 제공자는 CreatorSymId를 갖게 됩니다.



# SCT 처리 방식

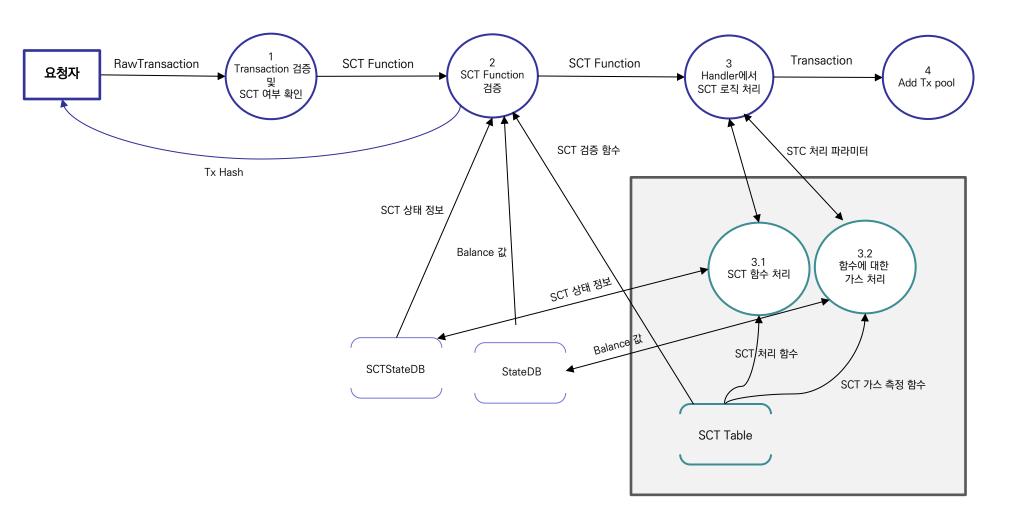
11

- 1) SCT 처리 프로세스
- 2) SCT 자료 구조



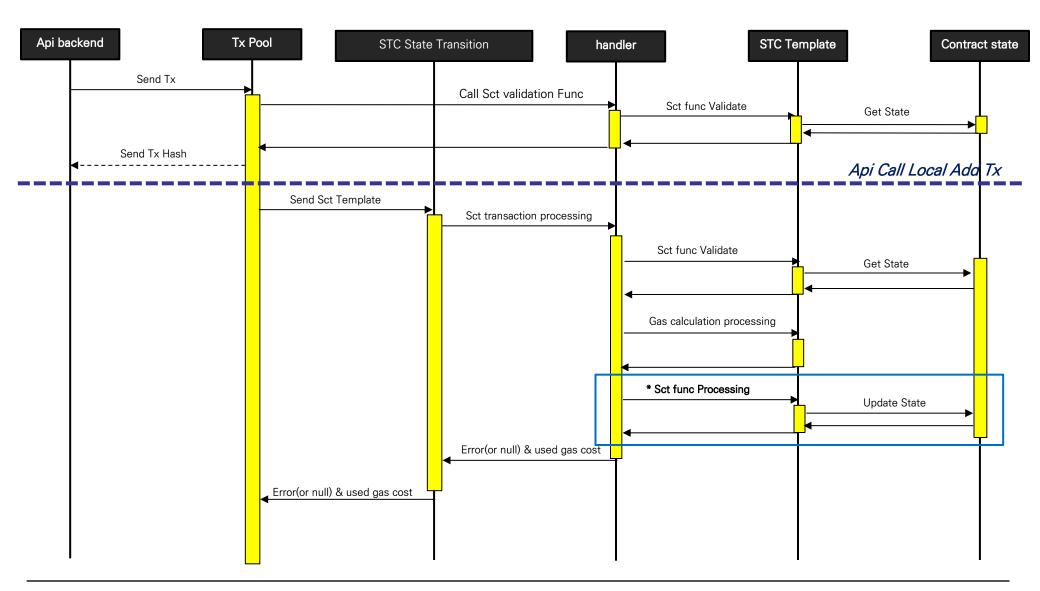
#### ❖ SCT 처리 프로세스

Gsym engine 내부에서 SCT Transaction 처리 과정입니다.



12





13



- 1. Type의 값이 1이고 Payload 의 값이 존재 할 경우 SCT 처리 프로세스 진행
- 2. Payload 값을 RLP Decoding 하여 SCT-FUNC를 생성 및 검증
  - 2.1 RLP Decoding 후 정상적으로 SCT-FUNC 생성했는지 체크
  - 2.2 SCT-FUNC 를 이용하여 SCT-FUNC-TABLE에서 Template의 <u>HUG 가격 계산 함수</u>와 <u>검증 함수</u>를 통과하는지 체크
- 3. SctFuncHanlder를 통해 SCT-FUNC-TABLE 에서 매칭되는 Template 의 <u>가스 가격 계산 함수</u> 및 <u>처리 함수</u>를 실행하고 그에 따른 상태 변경 값을 StateDB와 SctStateDB 저장

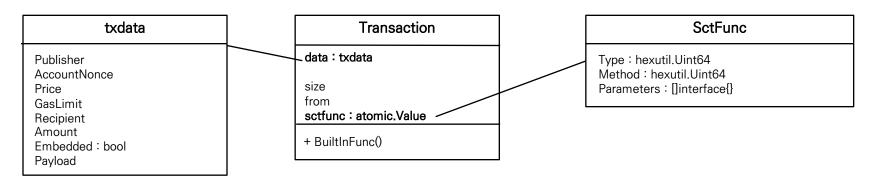
14

4. 그 후 SCT 처리 프로세스를 빠져나와 해당 Transaction 을 Block 기록 후 합의



#### ❖ SCT 자료구조

SCT 관련되어 있는 Transaction & State의 자료구조 입니다.

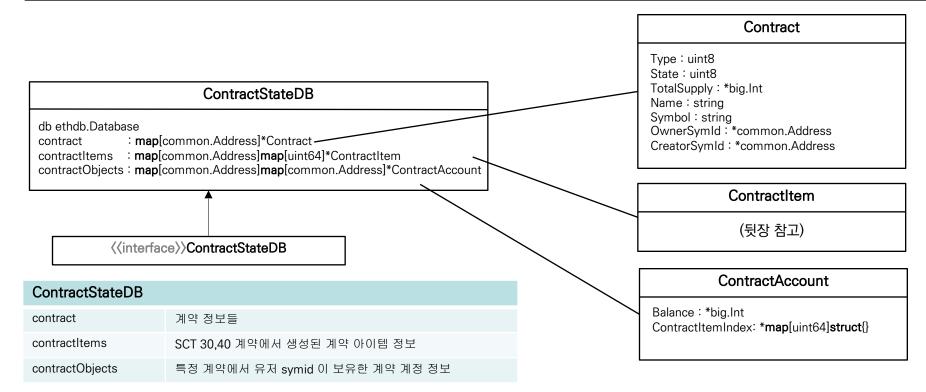


txdata	
type	트랜잭션의 타입 (1: sct )
Payload	SCT 트랜젝션일경우 Type, Method, BuiltInFuncParameterList 에 대한 RLP Encode 값

Transaction	
sctfunc	BuiltInFunc() 함수를 통해 저장된 StcFunc 의 임시 저장 값 ( cache )

SctFunc	
Туре	SCT 함수의 타입 예:(0: SCT20, 1: SCT30)
Method	SCT 함수의 처리 방식 예:(0: create, 1: transfer)
Parameters	SCT 함수의 처리에 필요한 변수 값들





Contract	
Туре	계약 타입 ( 0: SCT20, 1: SCT30, 2: SCT40 )
State	계약 상태 ( 0: active, 1: lock )
TotalSupply	SCT20 일 경우의 총 발행량 / SCT30, SCT40일 경우 ITEM 개수
Name	계약 이름
Symbol	계약 심볼
OwnerSymId	계약 소유자
CreatorSymId	계약 생성자

ContractAccount	
Balance	소유 양 (SCT20 일 경우)
ContractItemIndex	소요 아이텍 ( SCT20 30 익 경우 )





State: uint8

Type: []byte // 아이템 속성 Name: []byte // 아이템 이름 Value: big.int //아이템 가치 Category: []byte //아이템 카터

Category: []byte //아이템 카테고리
Property: []byte //아이템 고유 속성
Description: []byte //아이템 부가 설명

SCT30

State: uint8
Type: []byte
Name: []byte
Point: big.int
GroupNo: []byte
CouponNo: []byte
Etc: []byte

override

ContractItem		
State	아이템 상태	
Туре	아이템 타입	
Name	아이템 이름	
Value	아이템 가치	
Category	아이템 카테고리	
Property (*unique)	아이템 고유 속성	
Description	아이템 부가 설명	

CouponItem		
State	쿠폰 아이템 상태	
Type	쿠폰 타입	
Name	쿠폰 아이템 이름	
Point	쿠폰 아이템의 포인트 (해당 아이템의 가치)	
GroupNo	쿠폰 그룹 번호	
CouponNo (*unique)	쿠폰 시리얼 번호	
Etc	쿠폰 부가 설명	