BLOCKCHAINS

ARCHITECTURE, DESIGN AND USE CASES

SANDIP CHAKRABORTY
COMPUTER SCIENCE AND ENGINEERING,
IIT KHARAGPUR

PRAVEEN JAYACHANDRAN
IBM RESEARCH,
INDIA

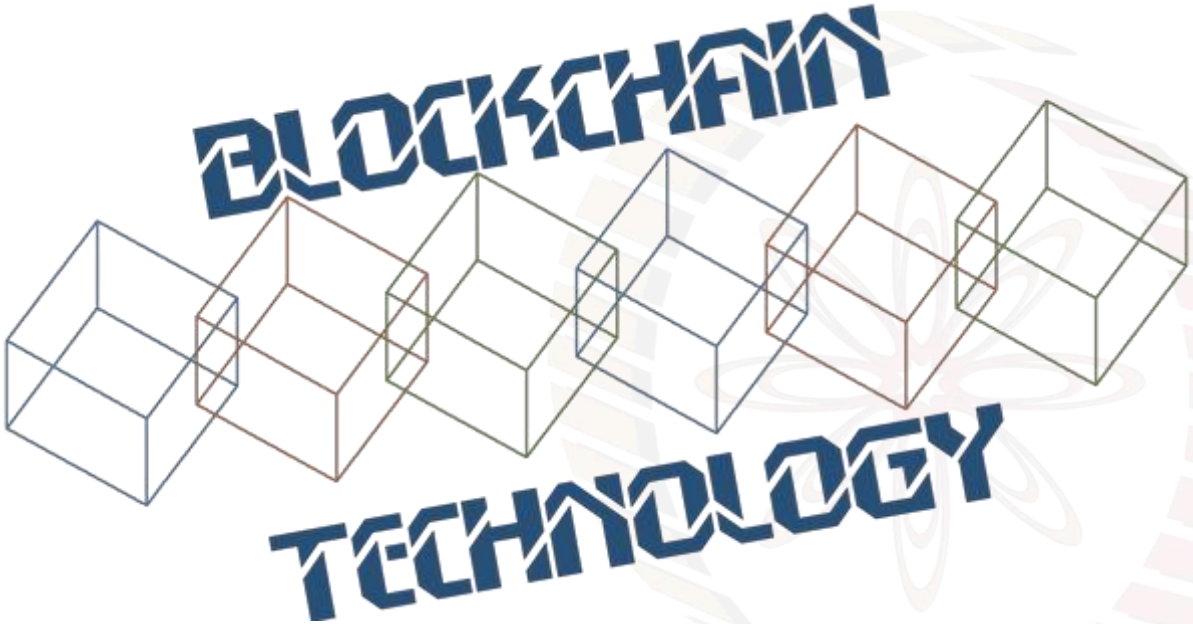
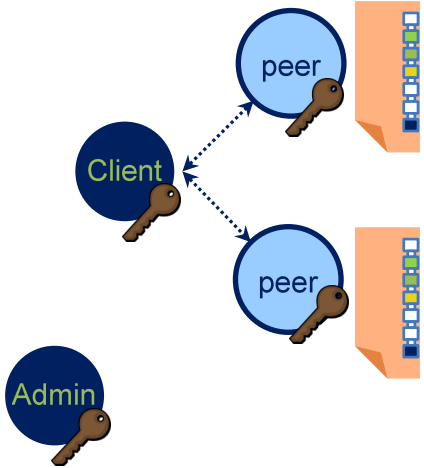
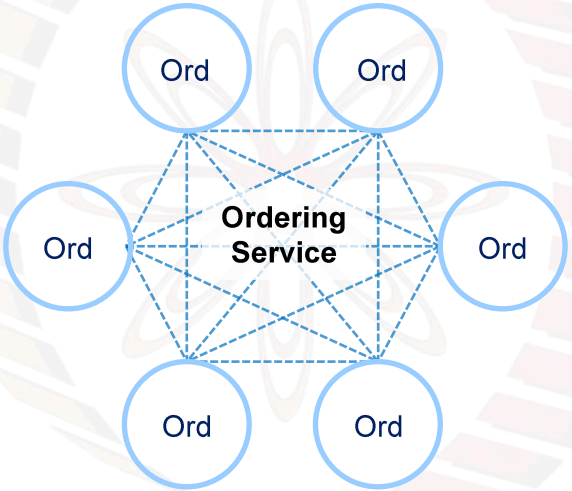


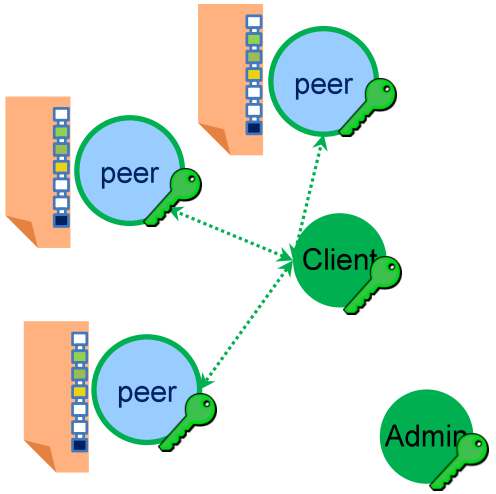
*Image courtesy: http://beetfusion.com/

FABRIC - MEMBERSHIP & ACCESS CONTROL

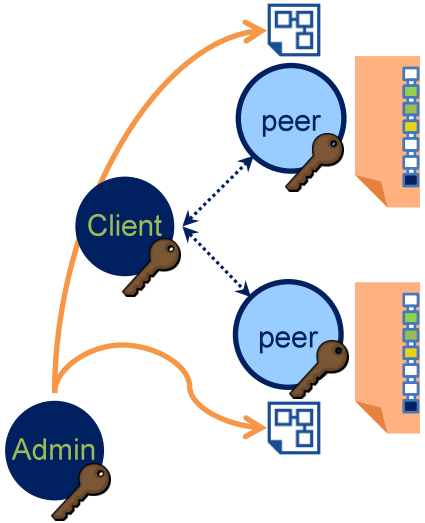
Identities and Policies Required at Every Stage

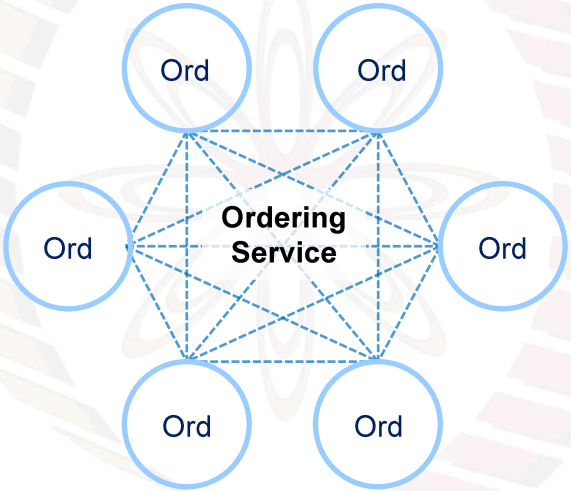


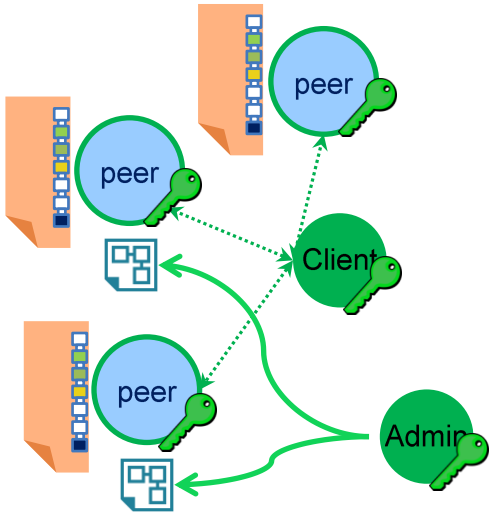




Identities and Policies Required at Every Stage

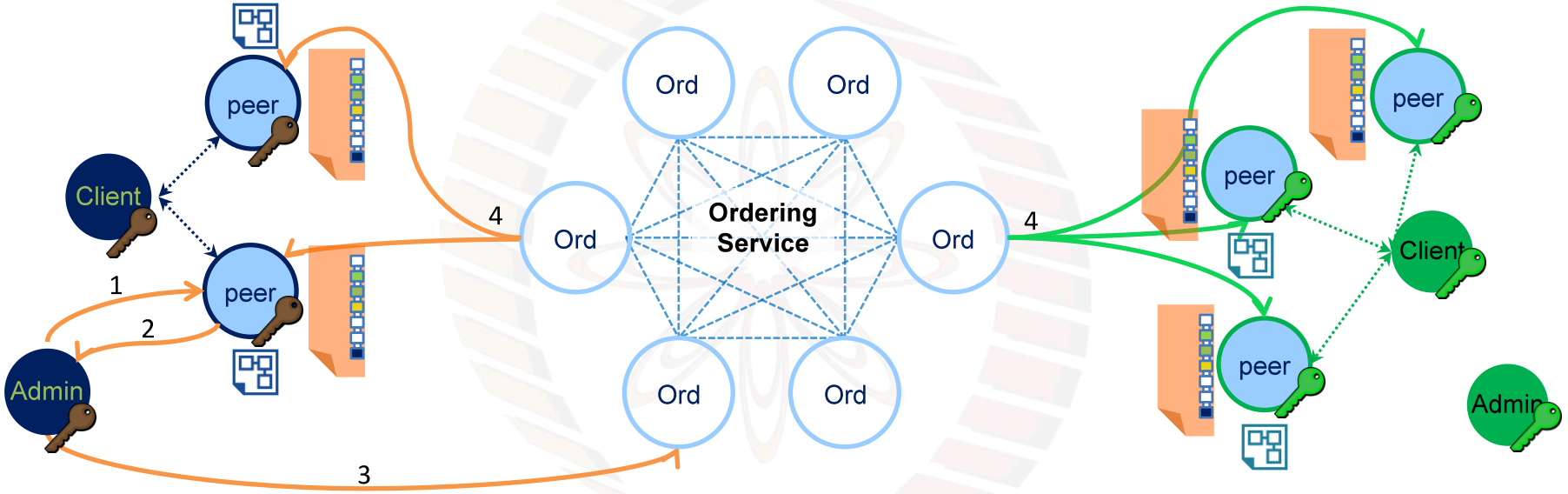






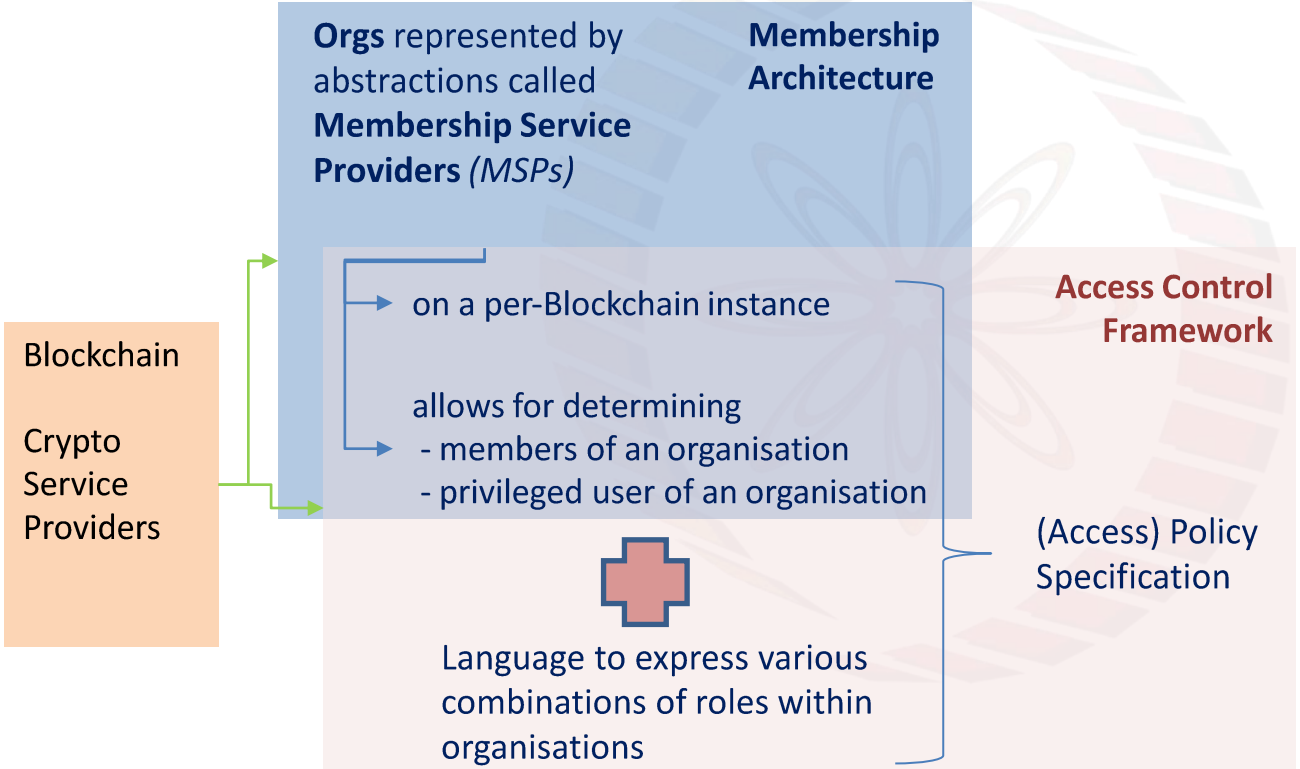
Installing chaincodes

Identities and Policies Required at Every Stage



Instantiating chaincodes
Invoking chaincodes
Reconfiguring the channel

Membership & Access Control Architecture



- System-level: Creation of new data partition (channels)
- Channel-level: Management of channel membership
- Chaincode-level: Management of chaincode availability, management of chaincode data access

MSPs defined on a per channel level:

✓ Enables secure interaction of entities who belong to different organisations, subjected to different management standard

MSP Details

- Described by a generic interface to account for:
  - User credential validation
  - User (anonymous but traceable) authentication: signature generation and verification
  - (optionally) User credential issue
- Examples:
  - providers that use and parse X.509 certificate extensions in a special way
  - providers with advanced crypto protocols to perform anonymous user-authentication
  - providers that issue multi-signature certificates
- Fabric core (v1.0):
  - Verifier MSPs used for client/peer/orderer signature verification run on a channel basis
  - Signer MSPs run locally and extend verifiers with singing capabilities
- Application MSP (v1.1)
  - Signer MSP run on the client side and offer attribute based signing capabilities
  - Verifier MSPs used by chaincode to extrapolate client attributes

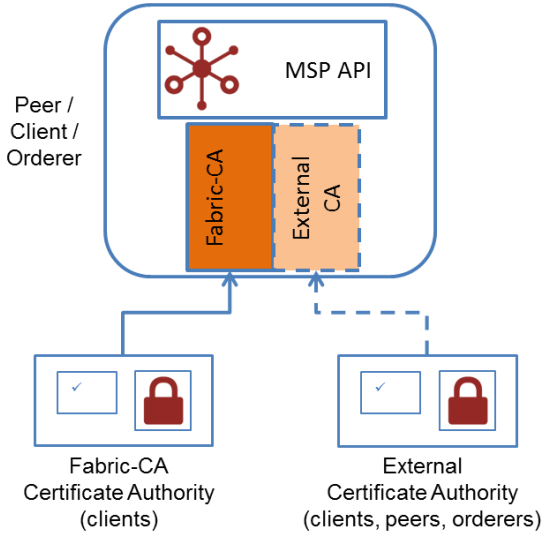
A Standard PKI based MSP for Fabric

VerifierMSP (on a per channel basis):

- Identity = standard X.509 certificate
- Governed by standard PKI hierarchies (root/intermediate CAs, CRLs)
  Setup = {list of root CAs, revocation list, admin certificate}
- Support for ECDSA keys/limited support for RSA keys
- Identity Validity Conditions = signed by a root CA
- Offers no anonymity or attribute support
- Signature generation/verification = standard public key crypto operation
- For certificate issuing, can leverage commercial CA (off-band), or our custom fabric-CA (online)
- Used by clients, peers, orderers for client/peer/orderer signature verification

SignerMSP (only on local basis):

- Verification aspects same as VerifierMSP
- Includes SigningIdentity = {stdrd X.509 certificate with public key PK, private key SK for PK}
- Used by clients, peers, orderers to sign messages & authenticate off-chain messages



MSPs: Building Blocks for Access Policies

Each MSP allows for the definition of three type of principals:

- Role-based: **member** of an MSP, **admin** of an MSP
- Identity-based: a specific **identity**
- Organizational-unit-based: a member of an MSP that belongs to a specific OU

A policy can be defined as a combination **N MSPPrincipals** and may **be satisfied** when identities and signatures corresponding to **t** of them are present

Examples: Assume the existence of three MSPs, **AliceCo**, **BobCo**, and **CharlieCo,** a policy can have the form

"AliceCo.admin **AND** CharlieCo.member"

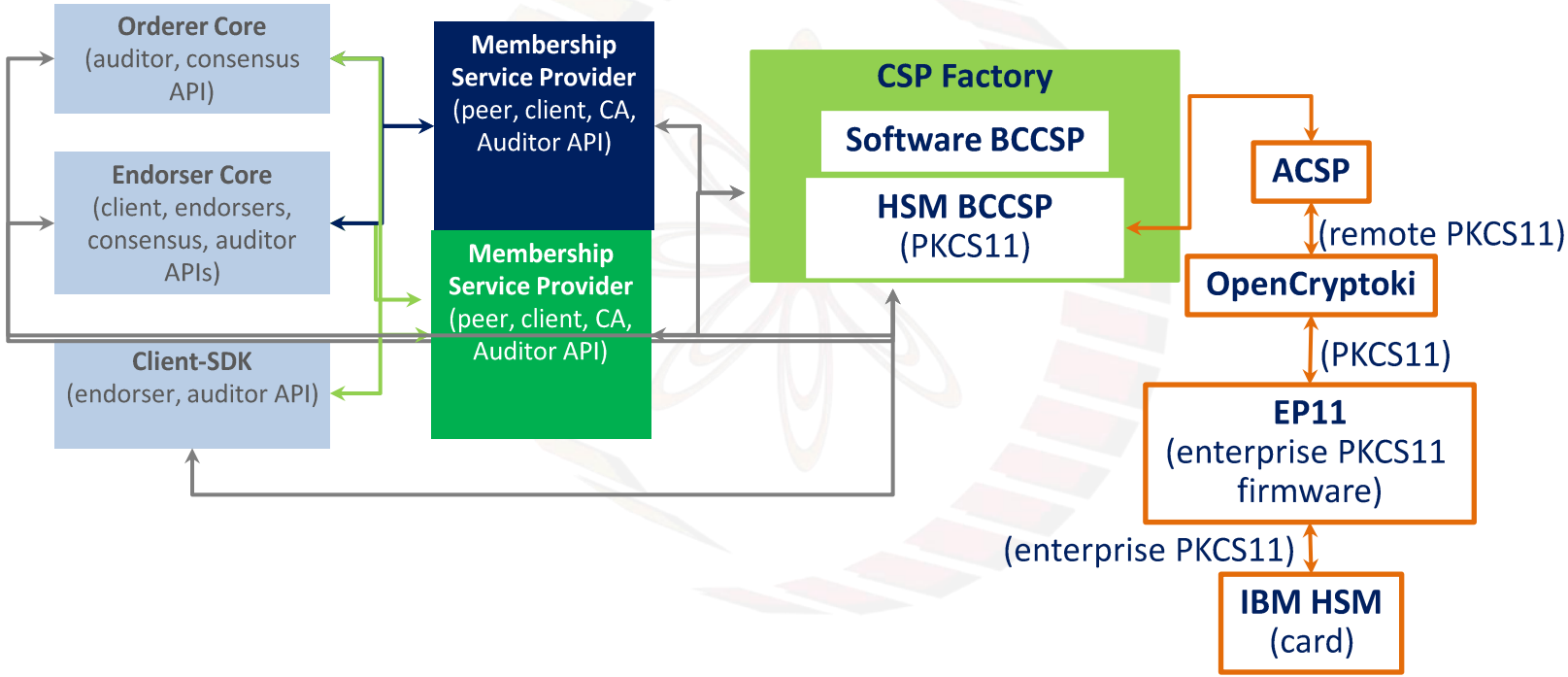
"AliceCo.admin **OR** CharlieCo.admin **OR** BobCo.admin"

"Charlie.OU.FinanceDivision"

Blockchain Crypto Service Providers (BCCSP)

	**Abstraction**	An abstraction of cryptographic operations used in Hyperledger Fabric
	**Pluggability**	Alternate implementations of crypto interface can be used within the HPL/fabric code, without modifying the core
	**Multiple BCCSP**	Easy addition of more types of CSPs, e.g., of different HSM types
	**International Standard Support**	Pluggable crypto service provider. Potential to support more fine-grained confidentiality features

Integration with HSM



Tool to Bootstrap a Network

- **Configtxgen (Configuration Transaction Generator)**
  - Network bootstrap tool
  - Designed to configure the network with organizations included in the ordering service genesis block and generates the configuration transaction artifacts used for channel creation (orderer, peers, CAs crypto material)

Fun Reading


thank you!