네트워크 기반 시스템 명세 및 검증을 위한 Z 프레임워크

신지훈, 최진영
고려대학교 컴퓨터학과
{jeehoon, choi}@formal.korea.ac.kr

요약

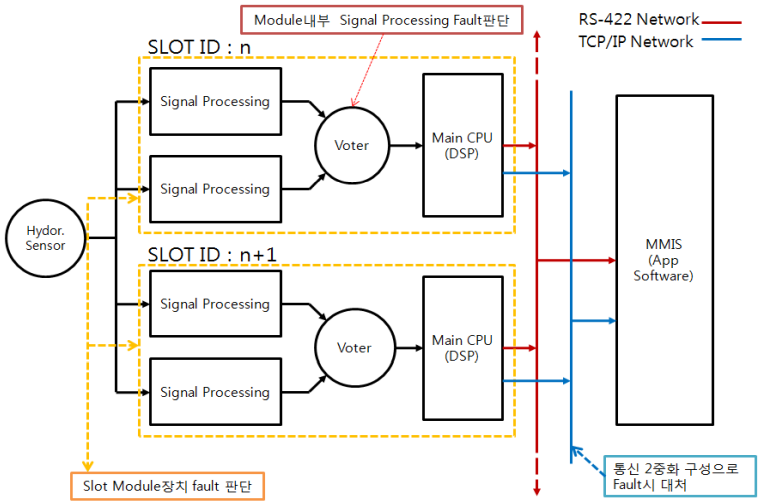
- 네트워크 환경의 시스템 기능 및 통신 명세(Z notation)
  - 네트워크 환경 명세
  - 통신 명세를 위한 템플릿 제시

- 사례연구(수소감시시스템)
  - 시스템 기능 및 통신 명세
  - 기능과 통신성이 결합된 속성 검증

관련연구

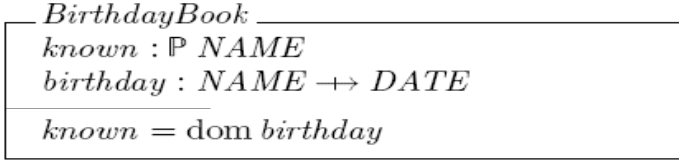
- 관련 연구
  - Specifying and Analyzing Security Automata using CSP-OZ
  - Formal Specification and validation of railway network components using Z notation

- 원자력 수소감시시스템
  - HMS, MMIS 기능
  - HMS ⇔ MMIS 통신

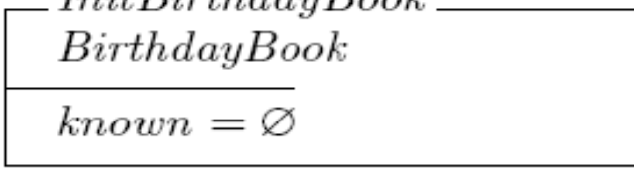


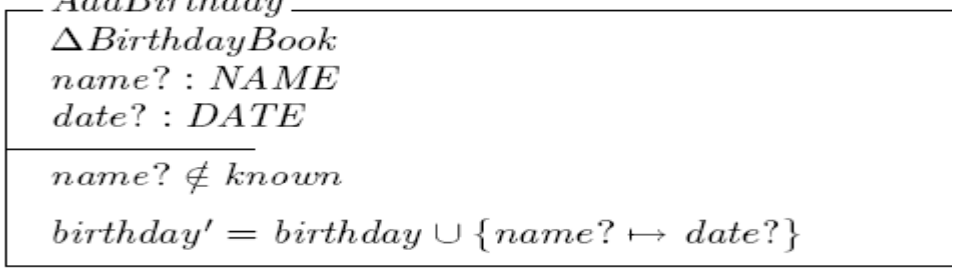
Z notation

- 집합론, 일차 수리논리에 기반을 둔 정형명세 언어
- Z 모델
  - 상태 모델 – 변수들의 선언, 변수들간의 값의 관계



  - 오퍼레이션 모델 – 초기 상태, 오퍼레이션 전과 후의 상태간의 관계

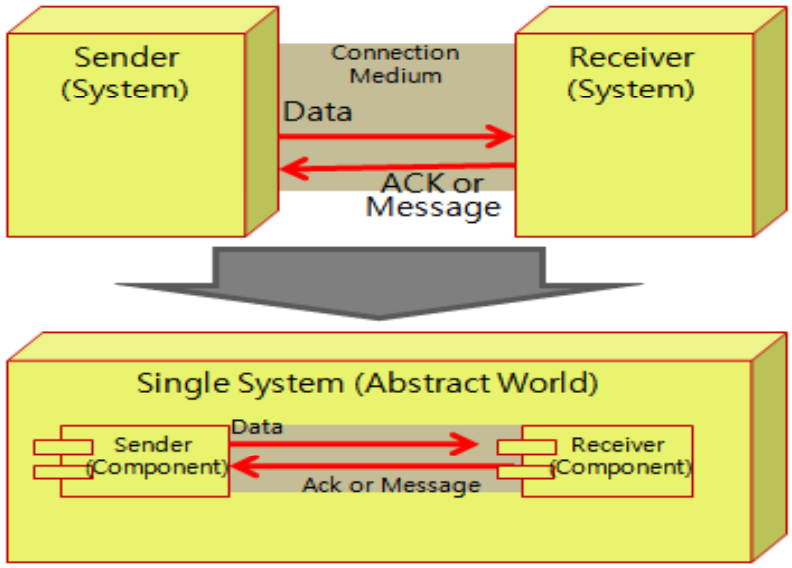




- Ex. Zen Project, Mondex Project, Correctness by Construction

Z 템플릿

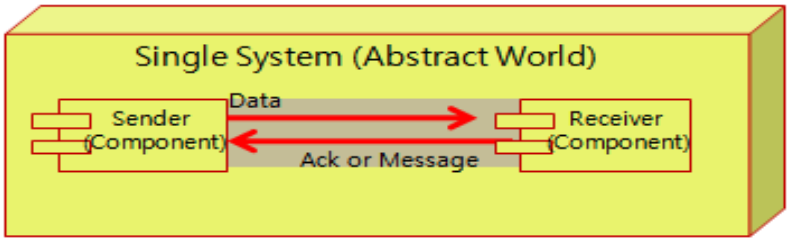
- 일반적인 네트워크 구성도와 데이터 전송 흐름도



  - 문제점
    - Z spec은 단일 시스템을 대상으로 함
    - 2개 이상의 시스템의 오퍼레이션들의 순서 및 관계를 나타낼 수 없음

Z 템플릿 명세

- Z 템플릿 명세



각 송신, 수신 시스템 및 커넥터를
단일 시스템(Abstract World)의 컴포넌트로써 명세

abstractWorld[SENDER, RECEIVER, PACKET]

sender: optional SENDER
receiver: optional RECEIVER
connector: Connector[PACKET]

Sender

data: DATA

Receiver

data: DATA
msg: Message

Connector[PACKET]

packets: ℙ PACKET

PACKET

packet_data: DATA
msg: Message

각 컴포넌트 간의 관계를 통해 데이터가 전송되는 것을 명세

데이터가 어떻게 구성되는 지, 데이터 전송 시 시스템들 간의 만족해야 할 사항들은 이 템플릿을 확장하면서 명세

sendingData_Base[S, R, P]

ΔabstractWorld[S, R, P]

sendingDataToConnector

sendingData_Base[Sender, Receiver, PACKET]

∃s: sender • connector'.packets = extractToPacket s.data
sender' = sender
receiver' = receiver

sendingDataToReceiver

sendingData_Base[Sender, Receiver, PACKET]

∃r: receiver' • abstractToData connector.packets = r.data
sender' = sender
connector'.packets = ∅

sendData ≙ sendingDataToConnector ⨾ sendingDataToReceiver

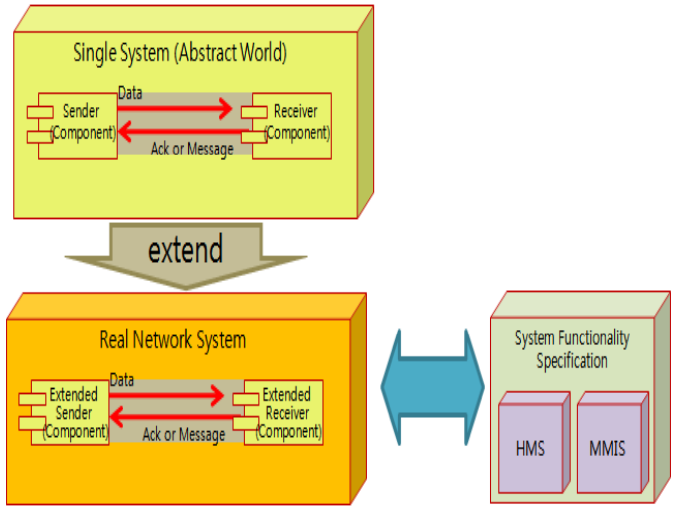
Response

ΔabstractWorld[Sender, Receiver, PACKET]

∃r: receiver • ∀packet: connector'.packets • packet.msg = r.msg
sender' = sender
receiver' = receiver

Z 템플릿 확장 명세

- Z 템플릿 확장 명세



기능 명세

HMS_State

$spm\_pairs: \mathbb{P}((SPM \times SPM) \times EquipID)$

$\#spm\_pairs = 2$

$\#(\mathrm{ran}\ spm\_pairs) = \#spm\_pairs$

SPM

$data: \mathrm{seq}\ IntegratedData$

IntegratedData

$high\_sense\_hydrogenation, low\_sense\_hydrogenation: \mathbb{N}$
$high\_sense\_temp, low\_sense\_temp: \mathbb{N}$
$high\_sense\_hydro, low\_sense\_hydro: \mathbb{N}$
$time: TIME$

$high\_sense\_hydrogenation = (high\_sense\_hydro - high\_sense\_temp) * 10$
$low\_sense\_hydrogenation = (low\_sense\_hydro - low\_sense\_temp) * 10$

measureSignalInSPM1

ΔHMS_State
$high\_sense\_hydro\_detection?, low\_sense\_hydro\_detection?: \mathbb{N}$
$high\_sense\_temp\_detection?, low\_sense\_temp\_detection?: \mathbb{N}$
$timeOfRTC?: TIME$
$tempDatum: IntegratedData$

IDInvariant

$tempDatum.high\_sense\_temp = high\_sense\_temp\_detection?$
$tempDatum.low\_sense\_temp = low\_sense\_temp\_detection?$
$tempDatum.high\_sense\_hydro = high\_sense\_hydro\_detection?$
$tempDatum.low\_sense\_hydro = low\_sense\_hydro\_detection?$
$tempDatum.high\_sense\_hydrogenation = (high\_sense\_hydro\_detection? - high\_sense\_temp\_detection?) * 10$
$tempDatum.low\_sense\_hydrogenation = (low\_sense\_hydro\_detection? - low\_sense\_temp\_detection?) * 10$
$tempDatum.time = timeOfRTC?$
$\forall spm\_pair: spm\_pairs; spm\_pair': spm\_pairs' \bullet$ **if** $spm\_pair.3 = id\_1 \land spm\_pair'.3 = spm\_pair.3$ **then** $spm\_pair'.1.data = (tempDatum)\ spm\_pair.1.data \land spm\_pair'.2.data = spm\_pair'.1.data$ **else** $spm\_pair' = spm\_pair$

통신 명세

HMS

Receiver

MMIS

Sender

HMS_MMIS_Protocol

$abstractWorld[MMIS, HMS, Packet]$

$\#sender = 1$
$\#receiver = 1$

Packet

PACKET
Command: CommandType
Preamble_code: seq *PreambleCodeType*
Id: ID
Checksum: YesNoType
Data_Size: $\mathbb{N}$

$\#Preamble\_code = 4$
$Data\_Size \geqslant 0$
$Data\_Size \leqslant 255$
$\#packet\_data = Data\_Size * 8$

real_sendingDataToConnector

real_sendingDataToConnector_Base

$\forall oneSender: sender \bullet \forall packet: extractToSystem\_Packet\ oneSender.data \bullet \#packet.Preamble\_code = 4 \land packet.Data\_Size \geqslant 0 \land packet.Data\_Size \leqslant 255 \land \#packet.packet\_data = packet.Data\_Size * 8$

real_sendingDataToConnector_Base

$\Delta HMS\_MMIS\_Protocol$
$sendingDataToConnector[templateConnector/connector, templateConnector'/connector']$

$connector = refineConnector\ templateConnector$
$connector' = refineConnector\ templateConnector'$

Z 템플릿 확장 명세 검증

- 기능 명세와 통신 명세간 매핑

기능 명세와 통신 명세 내의
시스템 상태 간의 매핑

HMS_relation

HMS
HMS_State

$spm\_pairs = dataRefine\ data$

$dataRefine: DATA \rightarrowtail \mathbb{P}\ (SPM \times SPM \times EquipID)$

$\#\ (\mathbf{ran}\ dataRefine) = 2$

매핑을 위한 함수 정의

기능 및 통신 속성 명세

$measureSignals \mathrel{\widehat=} measureSignalInSPM1 \wedge measureSignalInSPM2$

$CorrectOperationScenario \mathrel{\widehat=}$
$measureSignals \fatsemi compairHydrogenation \fatsemi real\_sendData$

theorem *CorrectFunctionCommunication*
$CorrectOperationScenario$
$\Rightarrow (\exists rcv: receiver' \bullet (\forall rcvData: dataRefine\ rcv\ .\ data \bullet rcvData \in SPM \times SPM \times EquipID$
$\wedge (head\ rcvData\ .\ 1\ .\ data)\ .\ high\_sense\_hydrogenation = (high\_sense\_hydro\_detection? - high\_sense\_temp\_detection?) * 10$
$\wedge (head\ rcvData\ .\ 1\ .\ data)\ .\ low\_sense\_hydrogenation = (low\_sense\_hydro\_detection? - low\_sense\_temp\_detection?) * 10))$

결론

- 정리
  - Z를 이용한 프로토콜 명세 프레임워크 제시
  - 템플릿을 확장한 실 시스템의 명세 및 검증
    - 네트워크 환경의 안전보안 필수 시스템 개발 대한 가이드 제시
- 향후 연구
  - Correctness by Construction 개발방법론 적용
    - 개발된 코드가 명세된 속성을 만족함
    - 제시한 프레임워크가 실제 시스템에 적용 가능함