

[*Espresso-Fastar* Workshop, Schanskop/Pretoria, October 2007]

# A new CSP Operator for Partial Parallelism

**Strauss/Kourie/Gruner/Roggenbach/Watson**

[ submitted to: Information Processing Letters – awaiting Review ]

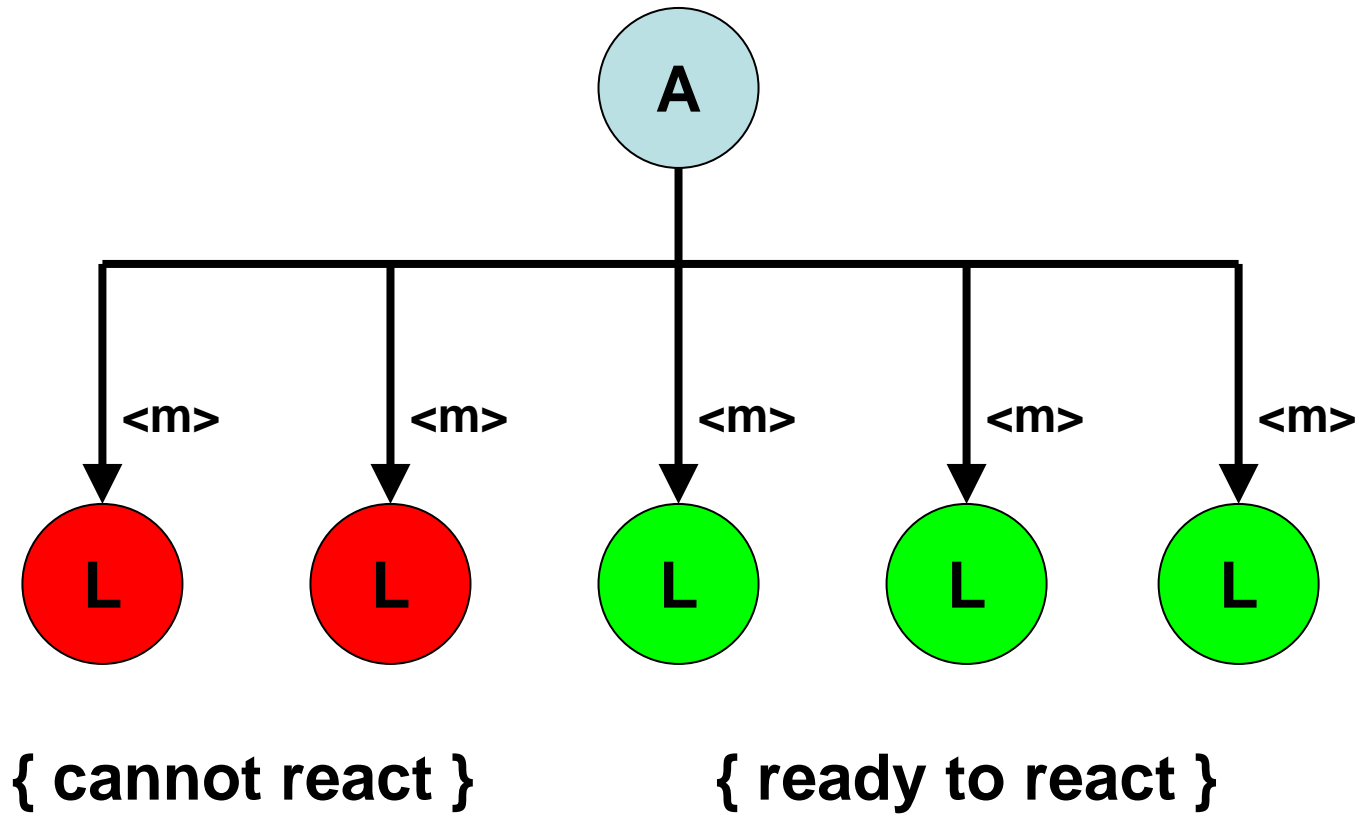
# Summary

- In our recent submission to the Information Processing Letters, we introduced a novel CSP Operator, characterized by *optional* (or partial) *Parallelism*
- We describe relevant application scenarios and provide the semantics of the operator in the basic trace-model  $\mathcal{T}$

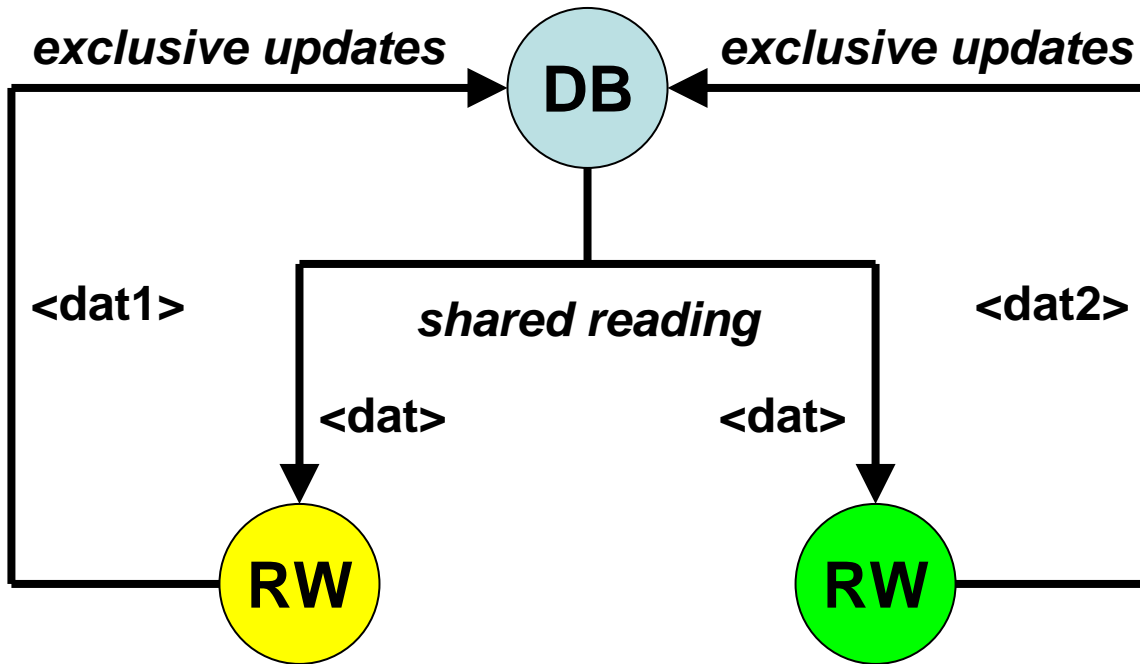
# Context of the Work

- Upcoming PhD Thesis: Tinus Strauss
- British/South-African Research Project: Roggenbach (GB) / Kourie-Gruner (ZA)
  - *Grant Acknowledgments:*
    - NRF
    - Royal Society

# Scenario (1): Broadcasting



# Scenario (2): Read-and-Update



# Comparison of CSP Operators

- Classical Parallelism Operator: “**all or nothing**”
- Either every Listener reacts, or the system does not progress
- Not suitable for modeling those scenarios

- **New Partial Parallelism Operator: “some may sleep”**
- System can make progress if at least one Listener can react
- Suitable for modeling those scenarios

# Comparison of CSP Operators

- The semantics of CSP Operators can be defined inductively in terms of Step-Rules
- Given two processes  $P_1$  and  $P_2$  and some Operator  $\Omega$ , the Step-Rule describes what can happen ***next*** in the combined process  $\mathbf{P} = (P_1 \ \Omega \ P_2)$
- Thereby it is also taken into account:
  - *what alphabet can be processed by  $P_1$*
  - *what alphabet can be processed by  $P_2$*
  - *what alphabet is allowed on their link-channel*

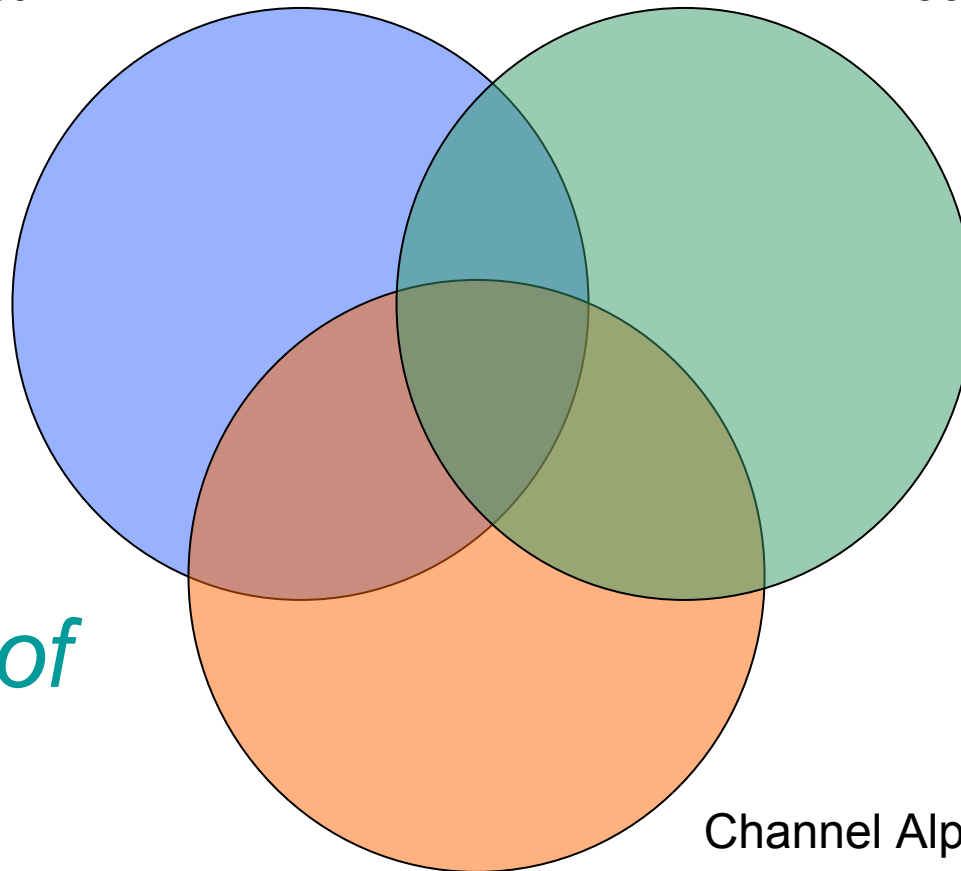
# Comparison of CSP Operators

- The formal description of the new Partial Parallel Operator (in terms of Step-Rules and Channel-Alphabets) reveals a **wider applicability of the new Partial Parallel Operator**, in comparison with the classical Parallel Operator.

# Comparison of CSP Operators

Process Alphabet A2

Process Alphabet A1



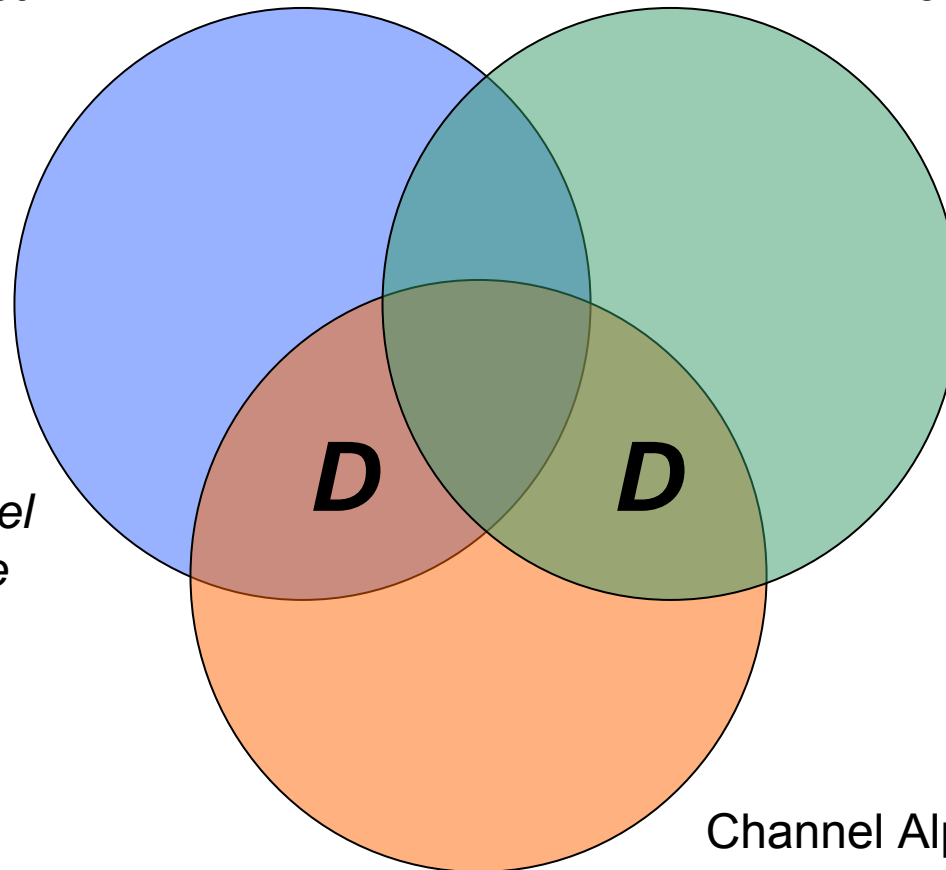
*Domain of  
Events*

Channel Alphabet X

# Comparison of CSP Operators

Process Alphabet A2

Process Alphabet A1



*Classical Parallel Operator, unlike the partial one, DEADLOCKS if event is from sub-domain **D***

Channel Alphabet X

# Properties of the new Operator

- The new Partial Parallel Operator is continuous in the Trace Model  $\mathcal{T}$
- Note:
  - Continuity is necessary for **Fixpoint** Proofs
  - The **Fixpoint** property is necessary for the validity of recursive process definitions with the new Operator, of kind  $\mathbf{P} = (\dots \Omega \dots \mathbf{P} \dots)$

# Properties of the new Operator

- **The new Partial Parallel Operator is also**
    - *idempotent*
    - *symmetric*
    - *associative*
- in the Trace Model  $\mathcal{T}$**

# Properties of the new Operator

- $P_1 \Omega\{\} P_2 = P_1 ||| P_2$ 
  - like classical on empty channel alphabet
- $P \Omega STOP = P$
- $P \Omega SKIP = P$
- $P_1 \Omega (P_2 \square P_3) = (P_1 \Omega P_2) \square (P_1 \Omega P_3)$
- Note:
  - A number of further properties can be found in our paper submitted to the Inform.Proc.Letters

# Possible Future Work

- More Proofs (*not all Properties yet proven*)
- More Properties (*only most obvious found*)
- Other Semantic Domains (*not only  $\mathcal{T}$* )
- Variants of the Operator (*e.g. parameters*)
- Experimental Validation and Tool-Support
- **Unified Theory** of Process Modeling (*?!?*)

# Time for Discussion...

## Any questions, suggestions, ideas...?