ThoughtWorks®

RISKA: Towards an Open-source RISC-V based Domain-specific System-on-Chip for SKA Data Processing

CARRV at ISCA 2021

17 June 2021

Arunkumar M. V., Harshal G. Hayatnagarkar
Engineering for Research, ThoughtWorks Technologies India

Square Kilometer Array

- **World's largest radio telescope** once completed - **International collaboration**
- Construction divided into **2 phases** Phase 1 - **SKA1** and Phase 2 - **SKA2**
- **SKA1** construction is slated to begin in the coming months

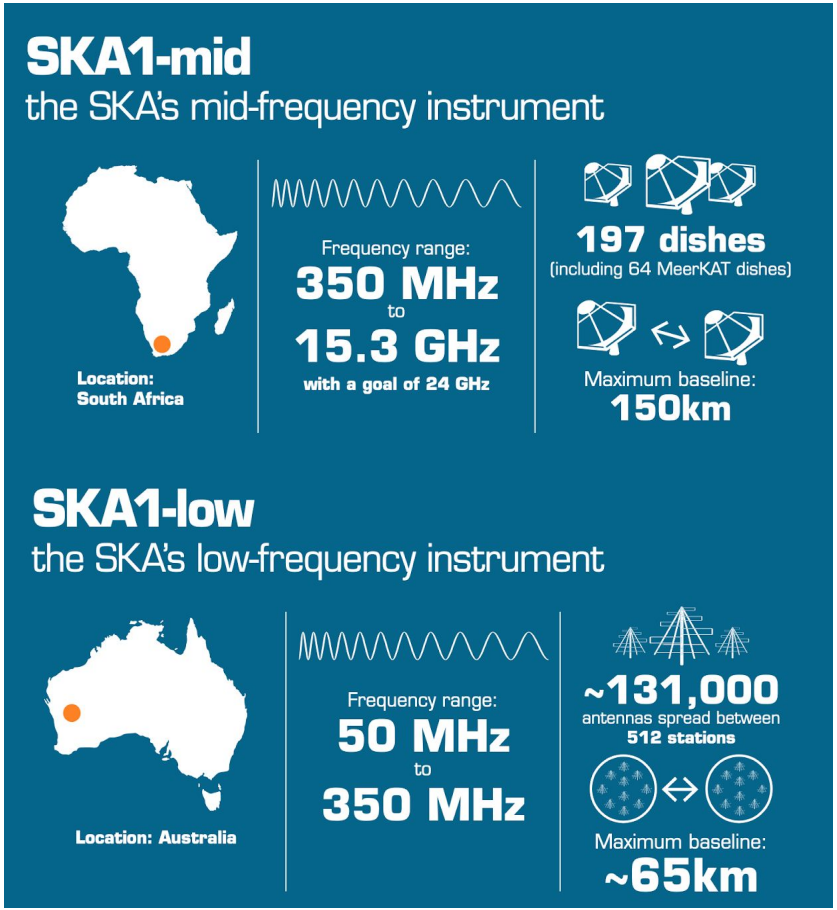


*Image Credit: SKA Info sheets – The telescopes, https://www.skatelescope.org/technical/info-sheets/

SKA: Big Data Challenge of the 2020s

- **SKA1**
  - **Data rate** of **~16 Tb/s** for SKA1
  - **Storage ~600 PB/year**
  - **Limited SDP energy budget ~2MW**
- **SKA2** will be much bigger than **SKA1 (~10x)**

<u>SKA2 Processing Challenge</u>

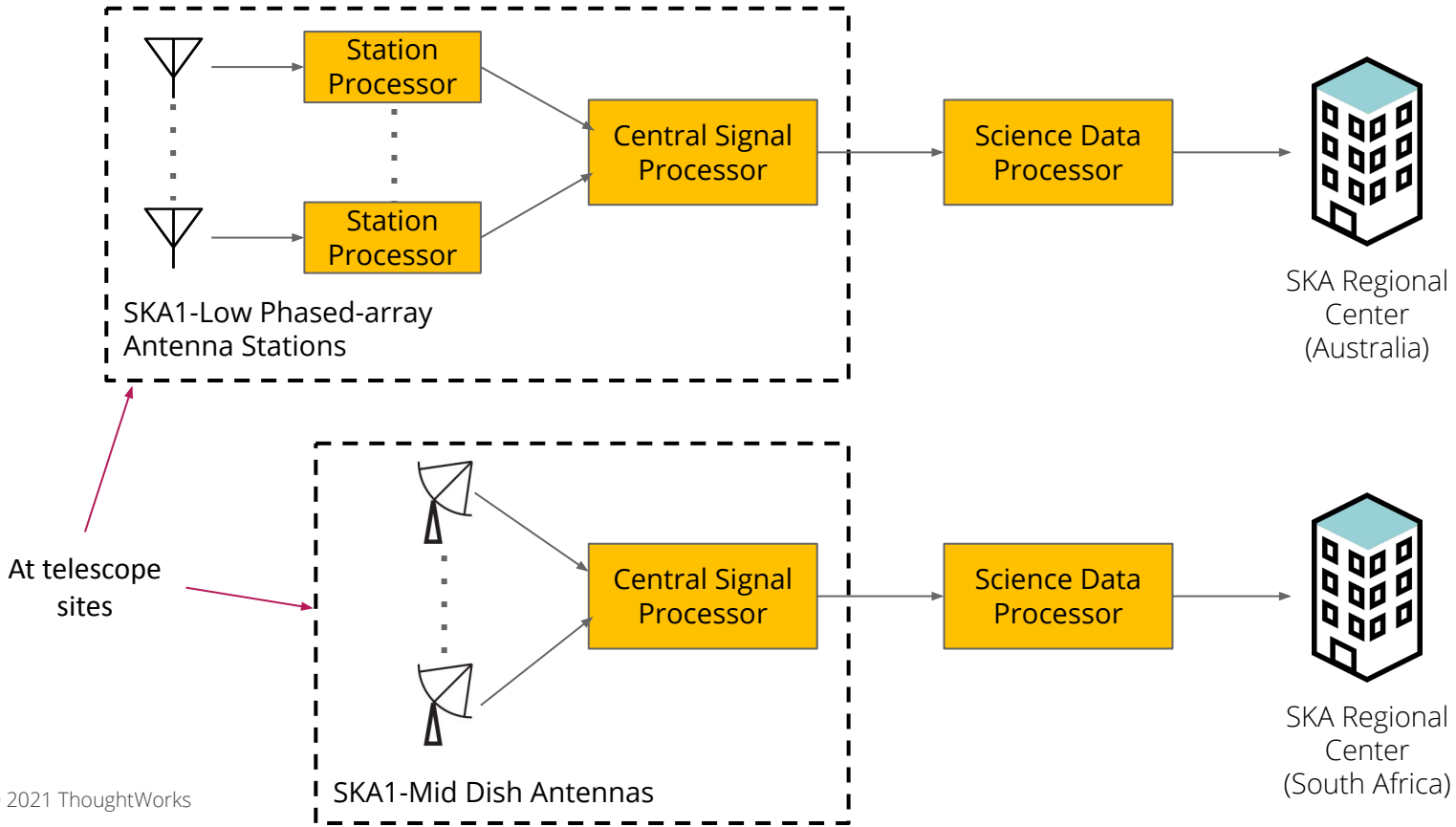
Exascale performance

at

Low energy consumption

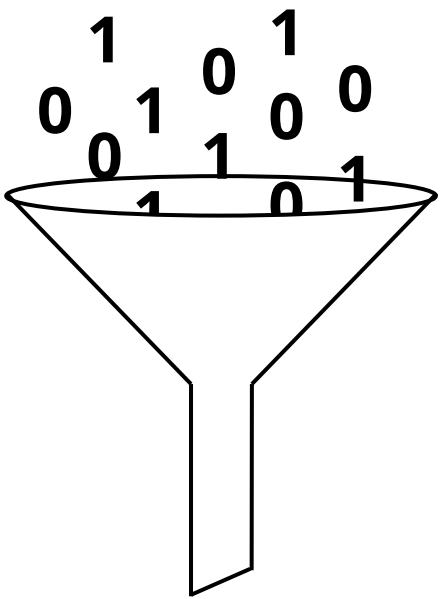
*Image Credit: SKA-mid Africa – Close up artist impression, https://www.skatelescope.org/multimedia/image/ska-mid-africa-close-up-artists-impression/

SKA processing pipeline

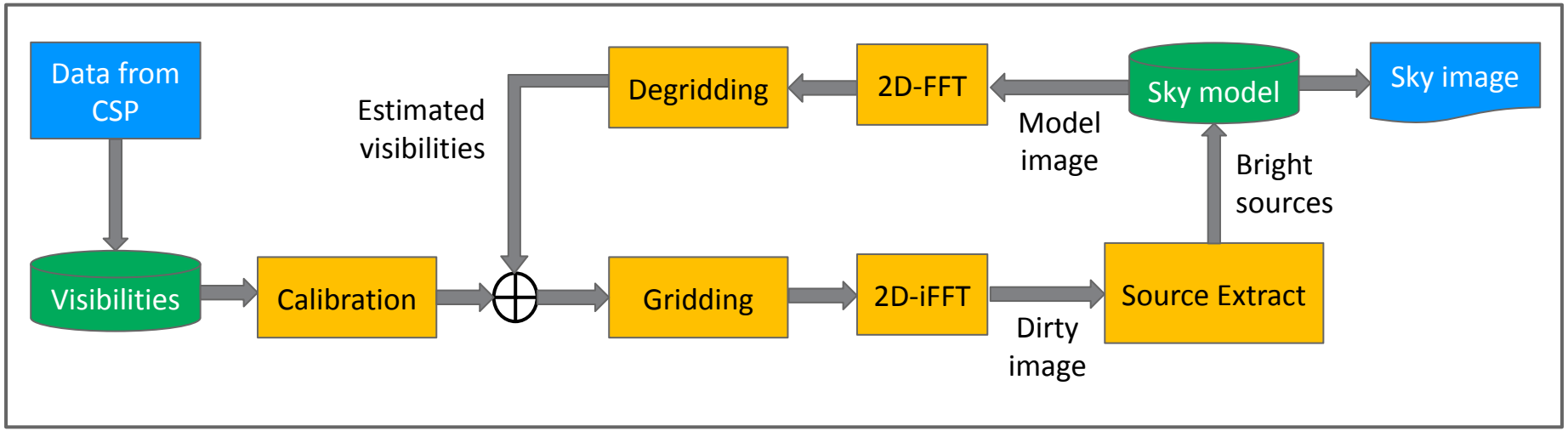


SKA Science Data Processor (SDP)

- **SDP transforms the observation data into the data products** for radio astronomers to use
- Enormous amounts of data have to be processed by the SDP **in near-real-time**
- **Imaging algorithms** including **gridding/degridding** and **2D-FFT** make up most of the computational requirements



SDP imaging pipeline



SDP: Science Data Processor
CSP: Central Signal Processor

Tile-based gridding algorithm

```
for all tiles in global grid do
    for all visibilities in tile do
        for v dimension in convolution kernel (that overlaps with tile) do
            for u dimension in convolution kernel (that overlaps with tile) do
                convolve visibility value with convolution kernel and add back to grid
```

SDP Bottlenecks and Remedies

Bottleneck	Remedy
Nature of algorithms	Choose appropriate algorithm for architecture
Low arithmetic intensity	Bring compute closer to data
Irregular Memory Accesses	Data reordering, larger caches
Off-chip Memory Bandwidth	On-chip High Bandwidth Memory
CPU Performance per Watt	Offload to domain-specific accelerators
Data Transfer to-and-fro Discrete Accelerators	Integrated accelerators
Memory Size in Discrete Accelerators	Integrated accelerators with uniform-memory access

These remedies point to two strategies ***Domain Specific Architectures*** and ***Integration.***

RISKA Hypothesis

DSAs specifically designed for the most time-consuming algorithms of SKA-SDP, and integrated with the CPU into a System-on-Chip (SoC) will address its overall challenges.

Domain Specific Architecture (DSA)

- **DSAs** are designed to **perform specialized computing tasks** - Do only some tasks but do them well
- Delivers **higher performance and energy efficiency** compared to CPUs / general purpose architectures (GPA)
- Requires **different programming models**
- Examples: GPU, Tensor Processing Units (TPU), Intel Crest etc.

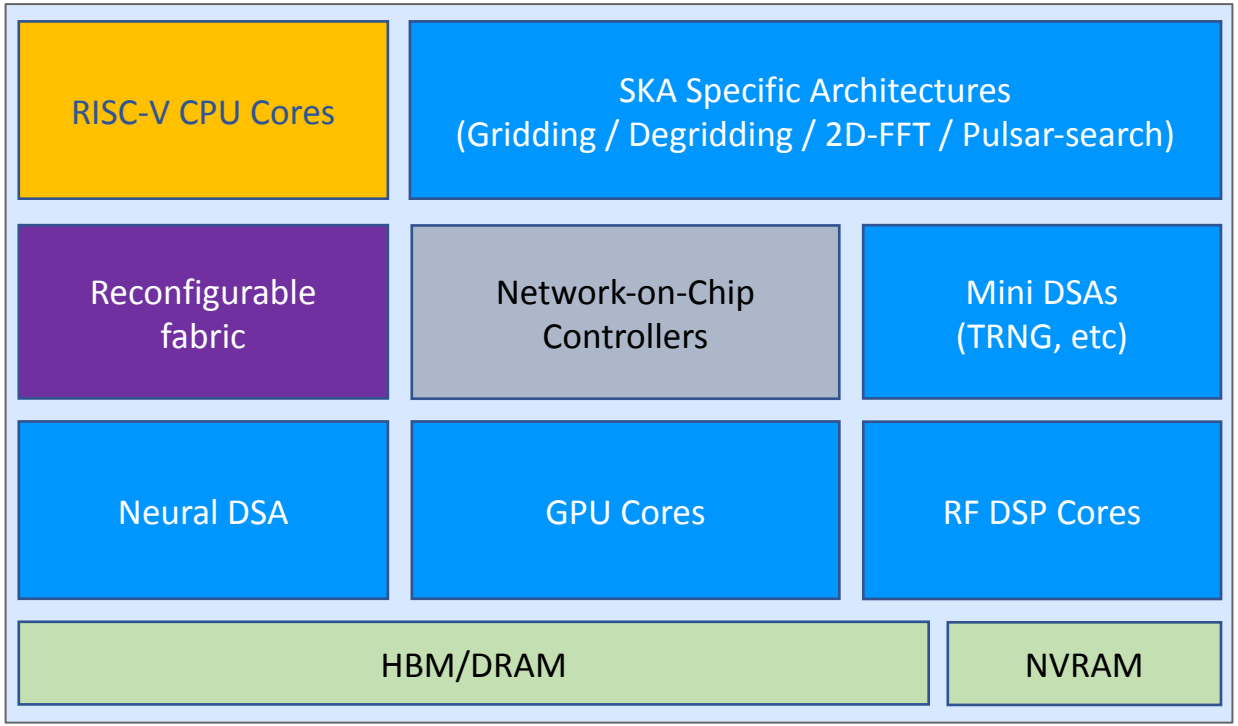
Why DSA ?

- DSAs will be able to achieve the performance and energy efficiency that SKA requires
- Implementations on both ASIC (concrete) and FPGA (reconfigurable) have shown **vastly better performance per watt** compared to implementations on CPUs
- Radio astronomy **imaging algorithms exhibit some form of data parallelism** which makes them suitable for DSA implementation

Why integrate DSA into SoCs ?

- **Higher interconnect performance**
  - PCIe bandwidth is a major inhibitor for discrete DSAs
  - Improved fine-grained communication
- **Uniform memory access** for DSAs
- **Locality** - Lower latency for data movement
- **Power efficiency** - Circuits in single die means less power for interconnect

Design Elements



RISC-V Cores

Why RISC-V would be the best fit for RISKA:

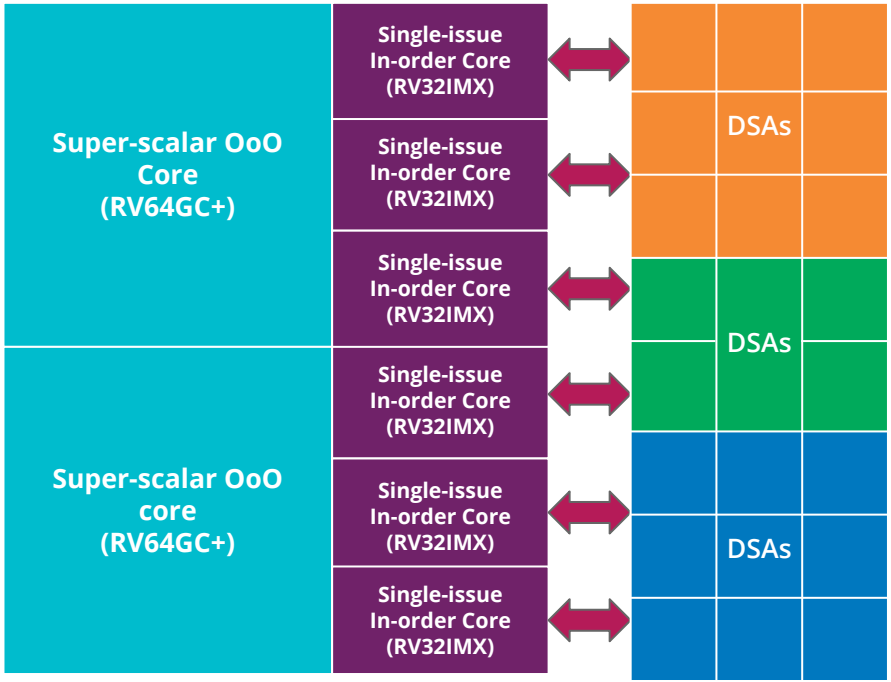
- **Free** & **open**
  - Possible to **develop in open source**
  - **Avoid vendor lock-in**

- **Minimal base** instruction set

- **Modular** & **extensible**

Many-core architecture

Enabled by the minimal base instruction set combined with the modularity & extensibility



RISC-V hurdles

- RISC-V is a **relatively young ISA** - “in its adolescence”
- Capabilities are yet to be ratified
- Some crucial ones in active development:
  - Vector extension
  - Cryptography extension
  - Debugging

“Whatever is broken/missing in RISC-V will get fixed” *

*Source: State of the Union - RISC-V Summit - December 2019

SKA specific architectures

Possible DSAs in RISKA:

- Novel accelerators
  - For **critical SDP imaging algorithms**
  - Domain specificity - can achieve **higher performance per watt**
  - **High programming complexity**

- Conventional accelerators
  - Architectures like **GPUs, Neural DSAs, RF DSPs and mini DSAs**
  - For applications where **satisfactory performance per watt** is achievable
  - **Established programming models**

Memory

- **Data locality** is a major issue with SKA data processing
- Bring **compute closer to data**
  - **In-memory** computing
  - **Near-memory** computing
- **High Bandwidth Memories** to speed up memory-bound algorithms
- Other novel strategies will also be necessary

Other design elements

- **Reconfigurable fabric**
  - **DSAs for all required algorithms may not be viable**
  - Reconfigurable architecture / **FPGAs integrated into the SoC**
  - Support for **evolving algorithms**

- **Network-on-Chip**
  - RISKA is an **accelerator-rich architecture**
  - SDP imaging algorithms are **data-intensive**
  - **On-chip communication** will become a **bottleneck**
  - NoC architectures may be the answer to this issue

Development and Evolution of RISKA

- Development of SoC is a mammoth task - **expensive and complex**

- To reduce cost
  - **Open-source model** of innovation
  - Leverage **free and open source tools**

- To reduce complexity
  - Effective **hardware-software codesign** approach is necessary
  - **Agile hardware development**

A **collaborative open source development model** will make RISKA viable

Conclusion

The SKA radio telescope once built will be the world's largest radio telescope and will generate unprecedented amounts of scientific data

The SKA SDP has to process the observation data in near-real-time and faces many architectural and algorithmic challenges

RISKA is our proposal for a domain-specific SoC to widen the bottlenecks faced by SKA SDP

***Domain Specific Architectures* & *Integration* to realize higher performance per watt**

The RISC-V ISA has been chosen for RISKA due to its many favourable features

We propose that RISKA be developed via a collaborative open-source model

Thank you!

Harshal
harshalh@thoughtworks.com

Arun
arunkumar.mv@thoughtworks.com

ThoughtWorks®