

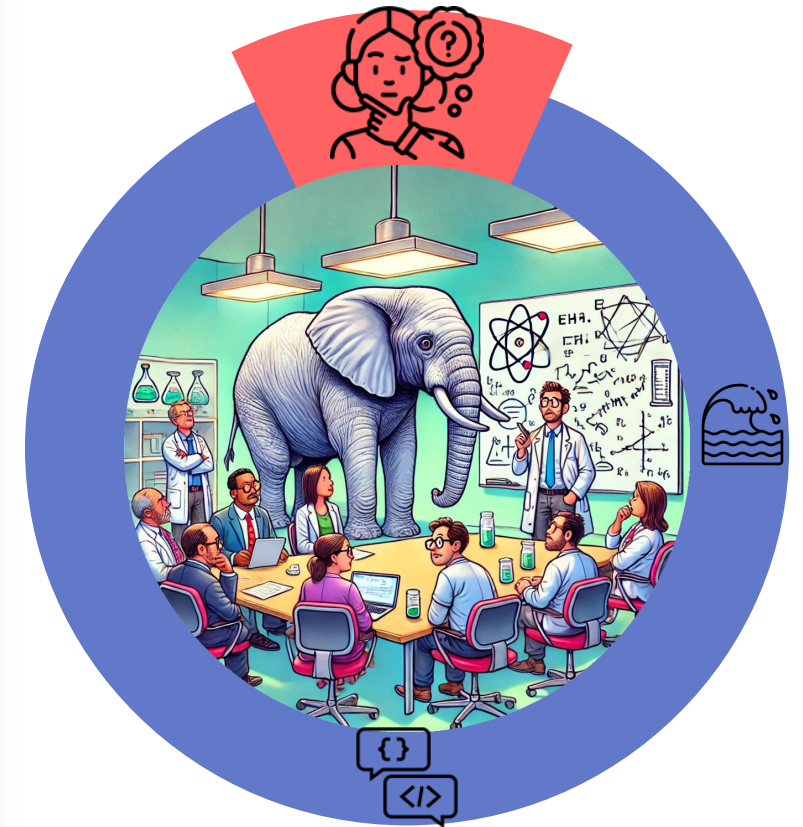


Breakout Group: YAC, ComIn and Beyond

When to use a coupler? When to use ComIn?

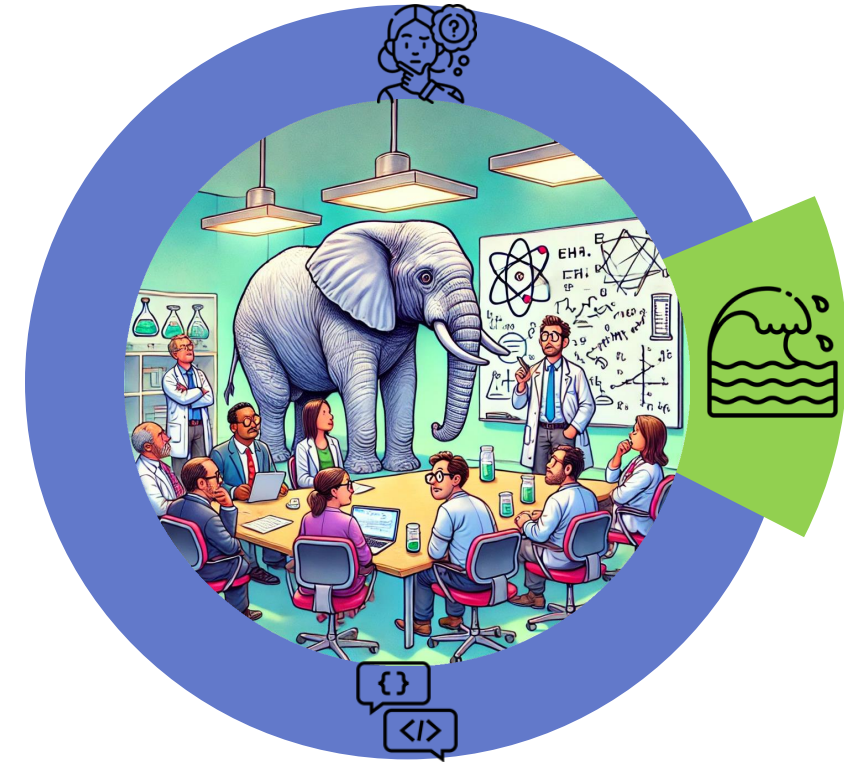
- **Choosing the right tool:**
 - Both, YAC and ComIn, have C/C++ and Python interfaces
 - Based on their requirements, projects should consider whether they need to use **YAC**, **ComIn** or **both of them**?

Are there cases that absolutely require implementation in ICON?



Interface design: Simple, stable & general = limited & restrictive?

- **Examples**
 - Granularity above the block loop level
 - Only global ICON variables are exposed
 - Inability to switch ICON processes on or off via plugins or coupled processes
 - ComIn: Descriptive data structures would lose their simplicity when in moving beyond the atmosphere component



Worst First:

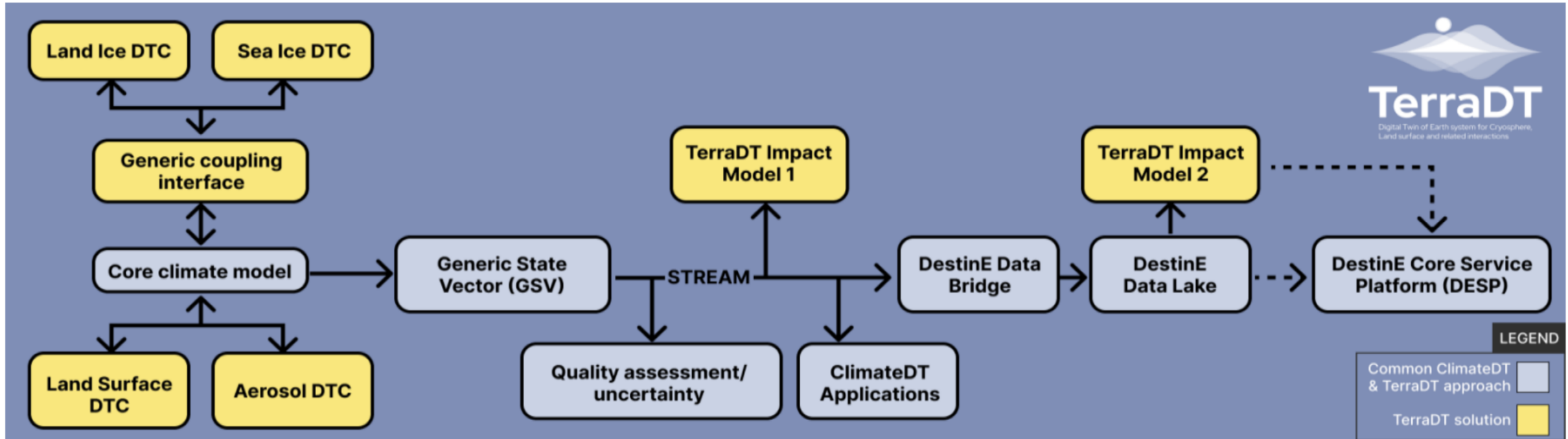
Where are natESM resources best invested?

- **Where work on interfaces is most needed?**
 - which programming languages?
 - which model components?
- **Should natESM take care of data products needed for offline (input/output) coupling?**



Backmatter

Other activities



- TerraDT introduces a modular approach with standard interfaces
- Enabling easy integration of new components and enhancing scalability, interoperability, and interactivity of the DestinE infrastructure
- Of course, different models or applications need different interfaces.