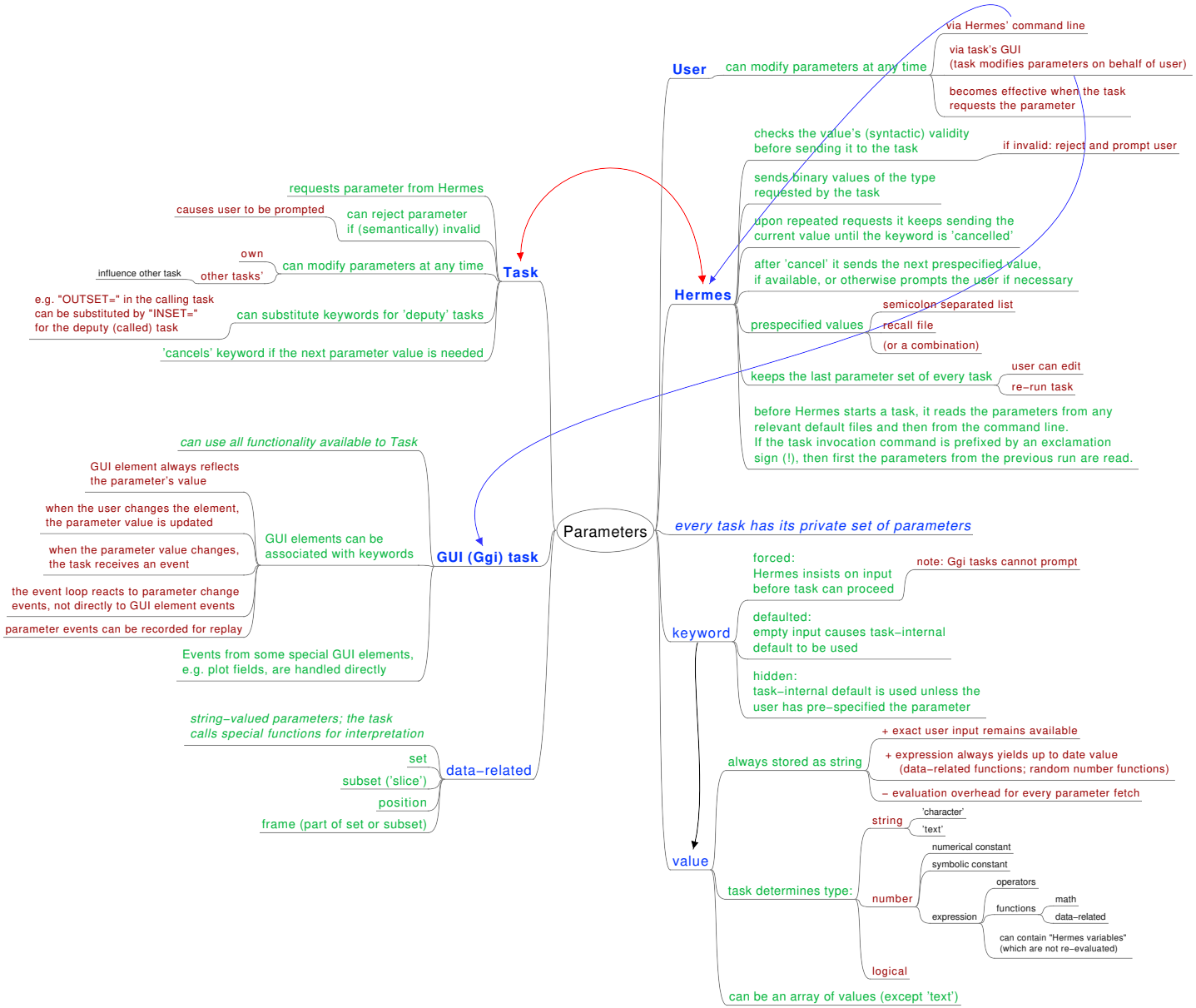
Parameters
User
can modify parameters at any time
via Hermes' command line
via task's GUI
(task modifies parameters on behalf of user)
becomes effective when the task
requests the parameter
Hermes
checks the value's (syntactic) validity
before sending it to the task
if invalid: reject and prompt user
sends binary values of the type
requested by the task
upon repeated requests it keeps sending the
current value until the keyword is 'cancelled'
after 'cancel' it sends the next prespecified value,
if available, or otherwise prompts the user if necessary
prespecified values
semicolon separated list
recall file
(or a combination)
keeps the last parameter set of every task
user can edit
re-run task
before Hermes starts a task, it reads the parameters from any
relevant default files and then from the command line.
If the task invocation command is prefixed by an exclamation
sign (!), then first the parameters from the previous run are read.
every task has its private set of parameters
keyword
forced:
Hermes insists on input
before task can proceed
note: Ggi tasks cannot prompt
defaulted:
empty input causes task-internal
default to be used
hidden:
task-internal default is used unless the
user has pre-specified the parameter
value
always stored as string
+ exact user input remains available
+ expression always yields up to date value
(data-related functions; random number functions)
- evaluation overhead for every parameter fetch
task determines type:
string
'character'
'text'
number
numerical constant
symbolic constant
expression
operators
functions
math
data-related
can contain "Hermes variables"
(which are not re-evaluated)
logical
can be an array of values (except 'text')
Task
requests parameter from Hermes
can reject parameter
if (semantically) invalid
causes user to be prompted
can modify parameters at any time
own
other tasks'
influence other task
can substitute keywords for 'deputy' tasks
e.g. "OUTSET=" in the calling task
can be substituted by "INSET="
for the deputy (called) task
'cancels' keyword if the next parameter value is needed
GUI (Ggi) task
can use all functionality available to Task
GUI elements can be
associated with keywords
GUI element always reflects
the parameter's value
when the user changes the element,
the parameter value is updated
when the parameter value changes,
the task receives an event
the event loop reacts to parameter change
events, not directly to GUI element events
parameter events can be recorded for replay
Events from some special GUI elements,
e.g. plot fields, are handled directly
data-related
string-valued parameters; the task
calls special functions for interpretation
set
subset ('slice')
position
frame (part of set or subset)