Contract Profiling

Version 6.1.1

November 4, 2014

`(require contract-profile)`   package: `contract-profile`

This module provides experimental support for contract profiling.

`(contract-profile body ...)`

Produces several reports about the performance costs related to contract checking in *body*. Each of these reports is printed to a separate file.

- *Cost Breakdown* (in `tmp-contract-profile-cost-breakdown.txt`): Displays the proportion of *body*'s running time that was spent checking contracts and breaks that time down by contract, then by contracted function and finally by caller for each contracted function.

- *Module Graph View* (in `tmp-contract-profile-module-graph.dot.pdf`): Shows a graph of modules (nodes) and the contract boundaries (edges) between them that were crossed while running *body*.

  The weight on each contract boundary edge corresponds to the time spent checking contracts applied at this boundary. Modules written in Typed Racket are displayed in green and untyped modules are displayed in red.

  These graphs are rendered using Graphviz. The Graphviz source is available in `tmp-contract-profile-module-graph.dot`. The rendered version of the graph is only available if the contract profiler can locate a Graphviz install.

- *Boundary View* (in `tmp-contract-profile-boundary-graph.dot.pdf`): Shows a detailed view of how contract checking costs are spread out across contracted functions, broken down by contract boundary.

  Contracted functions are shown as rectangular nodes colored according to the cost of checking their contracts. Edges represent function calls that cross contract boundaries

1

and cause contracts to be applied. These edges are extracted from profiling information, and therefore represent incomplete information. Because of this, the contract profiler sometimes cannot determine the callers of contracted functions. Non-contracted functions that call contracted functions across a boundary are shown as gray ellipsoid nodes. Nodes are clustered by module.

Each node reports its (non-contract-related) self time. In addition, contracted function nodes list the contract boundaries the function participates in, as well as the cost of checking the contracts associated with each boundary. For space reasons, full contracts are not displayed on the graph and are instead numbered. The mapping from numbers to contracts is found in `tmp-contract-profile-contract-key.txt`.

These graphs are rendered using Graphviz. The Graphviz source is available in `tmp-contract-profile-boundary-graph.dot`. The rendered version of the graph is only available if the contract profiler can locate a Graphviz install.

```
(contract-profile-thunk thunk) → any
  thunk : (-> any)
```

Like `contract-profile`, but as a function which takes a thunk to profile as argument.