Compiler Construction

∼ Binding times ∽

What is binding?

Binding refers to the process of linking
identifiers to objects

Very vague!

What does this mean?
We must go into details!

Binding Time

When a binding from a name to an
object is made.

Binding Time	Examples
language design	`if`
language implementation	data width
program writing	`foo`, `bar`
compilation	static objects, code
linkage	relative addresses
loading	shared objects
execution	heap objects

Static Binding vs. Dynamic Binding (1/2)

Static (early) Binding

The compiler statically (at compile time) associates an identifier to some definition

Dynamic (late) Binding

The compiler needs to add code that identifies the kind of object at runtime then matches the identifier with the definition (for example function call)

Static Binding vs. Dynamic Binding (2/2)

Roughly, flexibility and efficiency

- are mutually exclusive
- depend on binding time.

```
          binding-time
early ----------------------> late

Inflexibility             flexibility
efficiency                Inefficiency
```
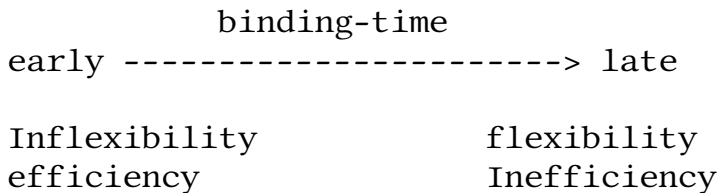
Dynamic Binding: `virtual` in C++

```cpp
struct Shape
{
  virtual void draw() const = 0;
};

struct Square : public Shape
{
  void draw() const override {};
};

struct Circle : public Shape
{
  void draw() const override {};
};
```

Dynamic Binding: **`virtual`** in C++

```cpp
#include <vector>
#include "shapes.hh"

using shapes_type = std::vector<Shape*>;

int main()
{
  auto ss = shapes_type{new Circle, new Square};

  for (auto s: ss)
    // Inclusion polymorphism.
    s->draw();
}
```

Very Late Code Binding: **eval** in Perl

```perl
sub try (&&@) {
  my ($try, $catch) = @_;
  eval { &$try };   # Explicit eval.
  if ($@) {
    local $_ = $@;
    &$catch;
  }
}
sub catch (&) {
  $_[0];            # implicit eval.
}

try {
  die "phooey";
} catch {
  /phooey/ and print "unphooey\n";
};
```

- Most interpreted languages support `eval` (explicit or not): runtime code evaluation.
- Enables language extensions.

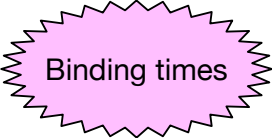# Binding Times in Tiger

Design | Keywords
Program | Identifiers
Compile | Function code, frames, types
Execution | Records, arrays addresses

> Little dynamic behavior!

Summary

Binding times

Dynamic Binding

Eval

Static Binding