Deceit_in_Detail_Tactic (marshaled)

```
entry /
  # Identify next marshal point
  # Load horse with arrows
  archer.arrows = HorseArcher.MAXIMUM_ARROW_CAPACITY
  archer.ticks = 0
  archer.post_fifo(Event(signal=signals.Second, times=0, period=archer.to_time(1.0), deferred=True)
second /
  archer.ticks += 1
exit / archer.cancel_event(Event(signal=signals.Seconed))
Senior_Advance_War_Cry as e\ archer.post_fifo(e)
Senior_Skirmish_War_Cry as e\ archer.post_fifo(e)
Senior_Retreat_War_Cry as e \ archer.post_fifo(e)
```

Advance

```
entry /
  chart.post_fifo(
    Event(signal=signals.Close_Enough_for_Circle),
      times=1,
      period=archer.to_time(3.0),
      deferred=True)

exit/ archer.cancel_events(
  Event(signal=signals.Close_Enough_for_Circle))

Senior_Advance_War_Cry / {}
Other_Advance_War_Cry / {}
```

Circle and Fire

```
second /
  if archer.ticks % 6 == 0:
    archer.arrow -= random.randint(1,3)
    if archer.arrows < 20:
      archer.post_fifo(
      Event(signal=signals.Skirmish_War_Cry))
  archer.ticks += 1
```

Close_Enough_For_Circle

Advance_War_Cry as e /
```
  archer.yell(e)
  for ip, other in archer.others.items():
    other.dispatch(e)
```

Other_Advance_War_Cry as e, ip: /
```
  archer.post_fifo(
    Event(
      signal=/
        signals.Advance_War_Cry))
  archer.other[ip].dispatch(e)
```

Skirmish

```
entry /
  archer.post_fifo(
    Event(signal=signals.Officer_Lured),
    times=1,
    period=archer.to_time(
      random.randint(40, 200)),
    deferred=True
  if archer.arrow < 10:
    archer.post_fifo(
      Event(
        signal=signals.Ammunition_Low))

Officer_Lured /
  chart.post_fifo(Retreat_War_Cry)

Senior_Skirmish_War_Cry / {}
Other_Skirmish_War_Cry / {}

second /
  if archer.tick % 3 == 0:
    if random.randint(1, 10) <= 4:
      archer.arrow -= 1
    if archer.arrows < 10:
      archer.post_fifo(
      Event(
        signal=signals.Ammunition_Low))
  archer.ticks += 1
```

```
exit /
  archer.cancel(Event(
    signal=signals.Ammunition_Low))
  archer.cancel(Event(
```

Officer_Lured /
```
  archer.scribble("Knight Charging")
  archer.post_fifo(
    Event(signal=signals.Retreate_Ready_War_Cry))
```

Ammunition_Low /
```
chart.post_fifo(
  Event(signal=signals.Retreat_Ready_War_Cry))
```

Retreat_Ready_War_Cry /
```
  ready = True
  for ip, other in mongol.others.items():
    if other.state_name != 'ead':
      ready &= other.state_name == 'Waiting'
  if ready:
    # let's make sure Gandbold isn't a chicken
    archer.post_fifo(
      Event(signal=signals.Retreat_War_Cry),
      times=1,
      period=archer.to_time(
        random.randint(10, 30)),
      deferred=True)
```

Waiting to Lure

```
entry /
  archer.scribble('put away bow')
  archer.scribble('pull scimitar')
  archer.scribble('act scared')

exit /
  archer.scribble('stash scimitar')
  archer.scribble('pull bow')
  archer.scribble('stop acting')

second /
  archer.ticks += 1
```

Other_Skirmish_War_Cry as e, ip \
```
  archer.post_fifo(
    Event(
      signal=/
        signals.Skirmish_War_Cry))
  mongol.other[ip].dispatch(e)
```

Skirmish_War_Cry /
```
  archer.yell(Skirmish_War_Cry)
```

Marshal

```
entry /
  archer.scribble("halt horse")
  archer.scribble("identify next marshal point")
  archer.scribble("field wrap wounds on self and horse")
  archer.scribble("drink water")
  chart.post_fifo(
    Event(signal=signals.READY),
      times=1,
      period=archer.to_time(3),
      deferred=True)
```

Ready (Marshal Point) /
```
  ready = True
  for ip, other in archer.others.items():
    if other.state_name != 'dead':
      ready &= other.state_name == 'waiting'
  if ready:
    archer.post_fifo(Advance_War_Cry)
```

Waiting to Advance

```
entry /
  mongol.post_fifo(
    Event(
      signal=signals.Advance_War_Cry),
    times=1,
    period=random.randint(30,120),
    deferred=True)
mongol.arrows = 60
```

Out_Of_Arrows

Feigned Retreat

```
entry /
  archer.scribble('fire on Knights')
  archer.scribble('fire on footman')
  if archer.arrows == 0
    archer.post_fifo(
      Event(signal=signals.Out_Of_Arrows),
      times=1,
      period=archer.to_time(random.randint(10, 30)),
      deferred=True)

exit /
  archer.cancel_events(
    Event(signal=signals.Out_Of_Arrows))
  archer.scribble("full gallop")

second /
  if archer.tick % 3 == 0:
    if random.randint(1, 10) <= 8:
    archer.arrow -= 1
    if archer.arrows == 0:
      archer.post_fifo(
      Event(
        signal=signals.Out_Of_Arrows))
  archer.ticks += 1
```

Retreat_War_Cry as e /
```
  archer.yell(e)
  for ip, other in mongol.others.items()
    other.dispatch(e)
```

Other_Retreat_War_Cry as e, ip:
```
  mongol.other[ip].dispatch(e)
```