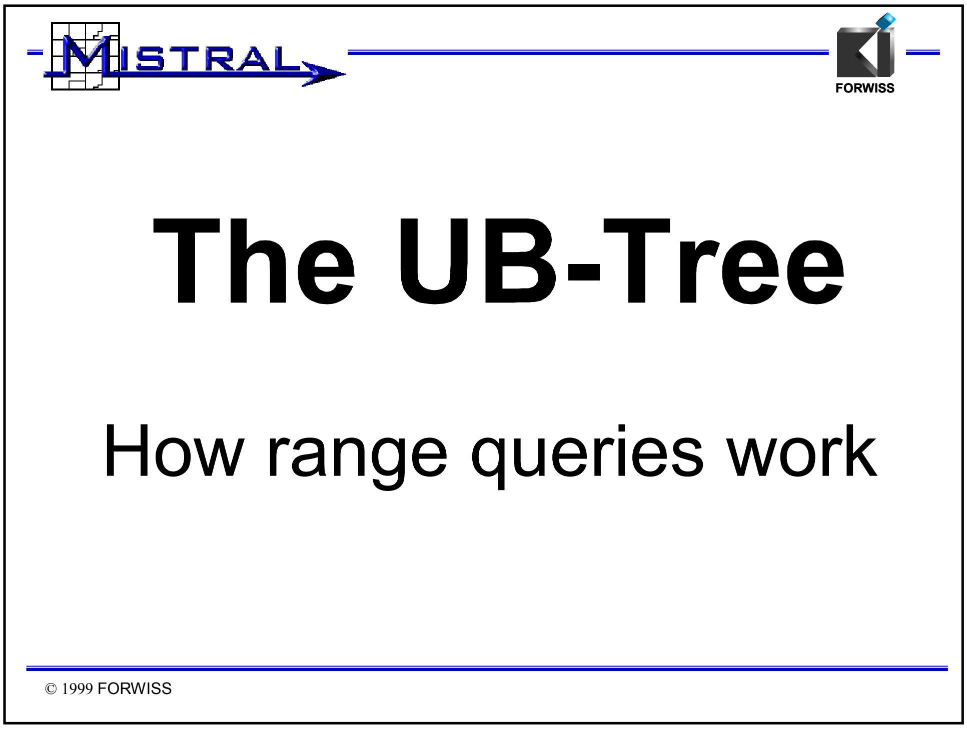
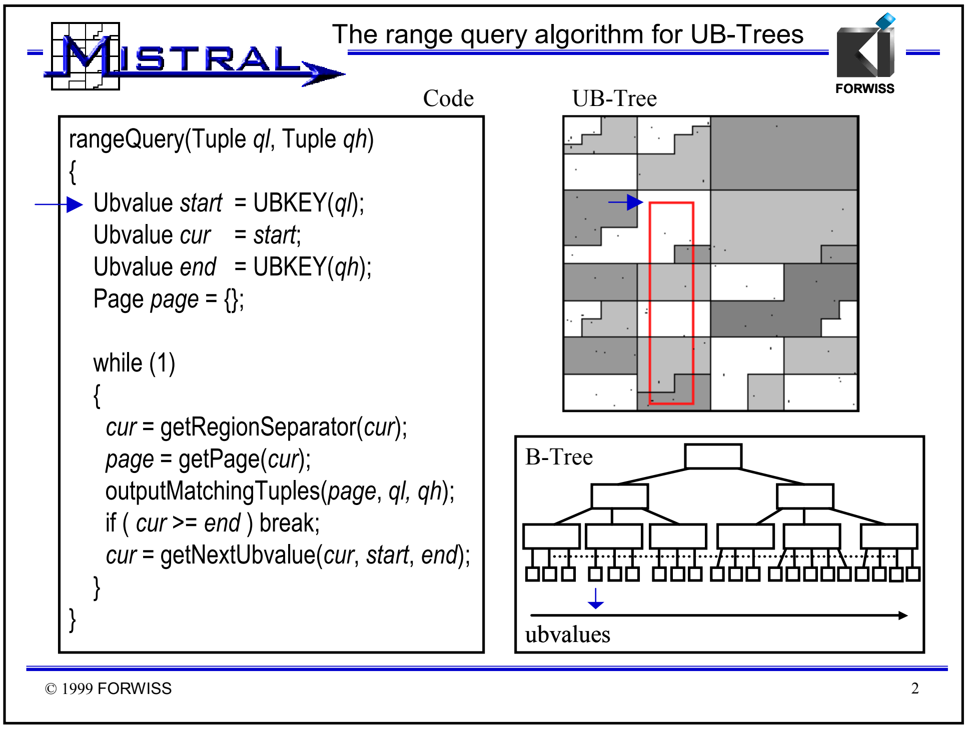


MISTRAL

FORWISS

The UB-Tree

How range queries work



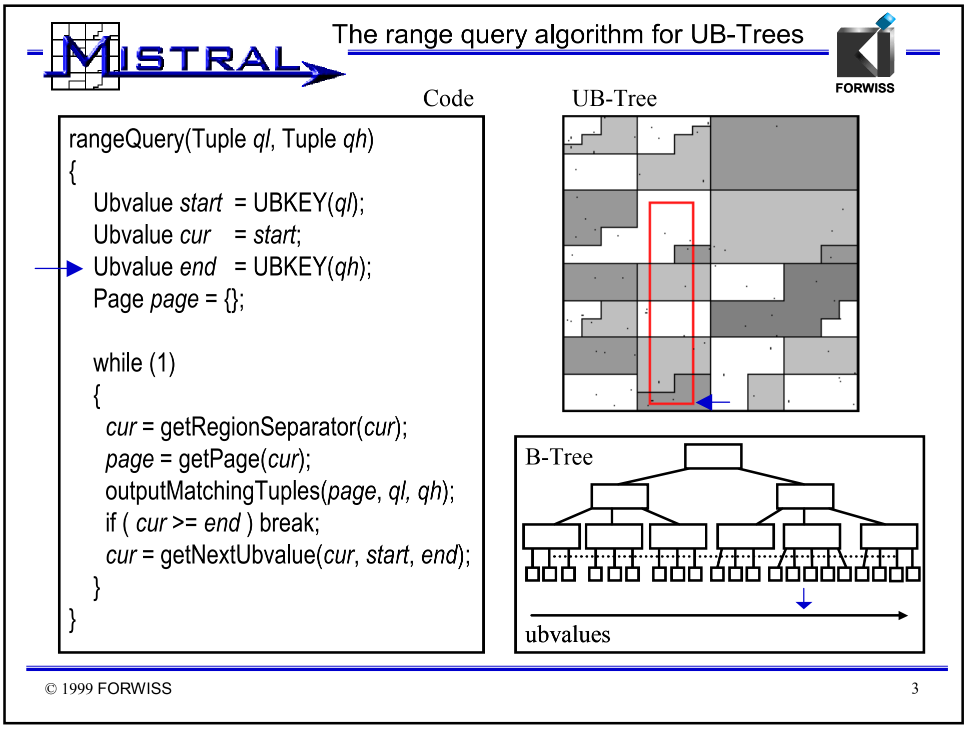
MISTRAL

The range query algorithm for UB-Trees

FORWISS

```
rangeQuery(Tuple ql, Tuple qh)
{
→ Ubvalue start  = UBKEY(ql);
  Ubvalue cur    = start;
  Ubvalue end    = UBKEY(qh);
  Page page = {};

  while (1)
  {
   cur = getRegionSeparator(cur);
   page = getPage(cur);
   outputMatchingTuples(page, ql, qh);
   if ( cur >= end ) break;
   cur = getNextUbvalue(cur, start, end);
  }
}
```

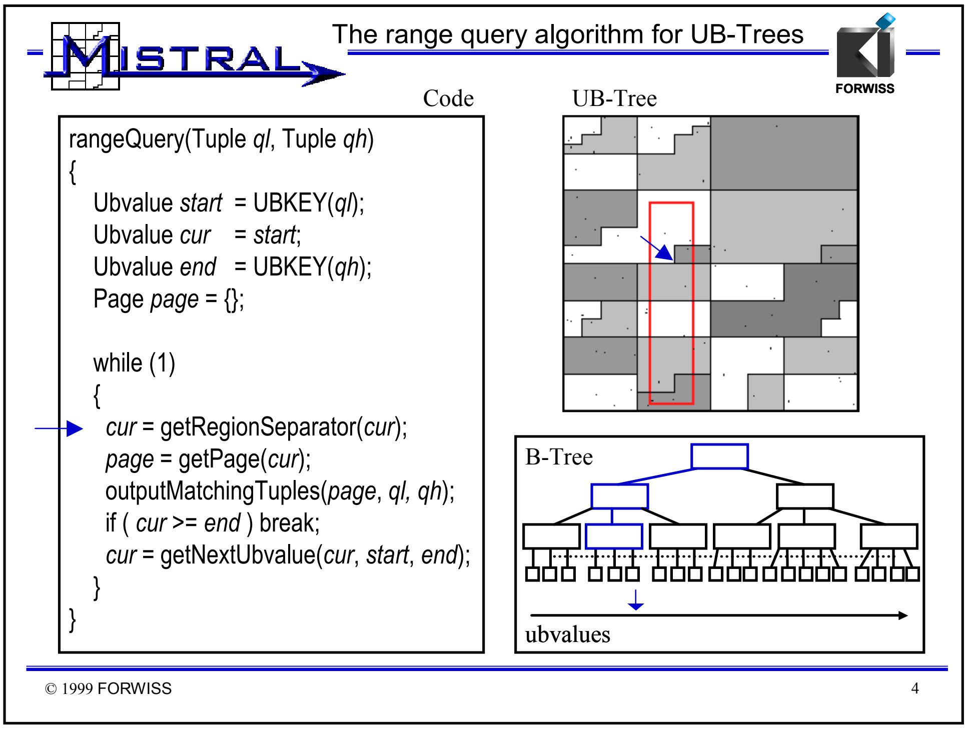


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

3

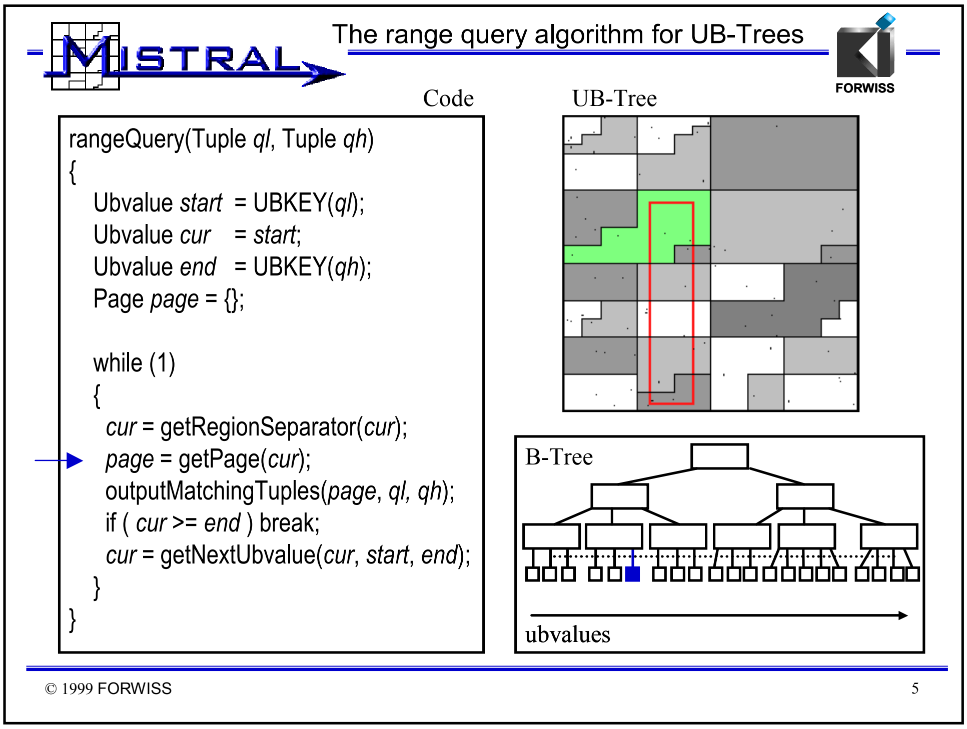


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

4



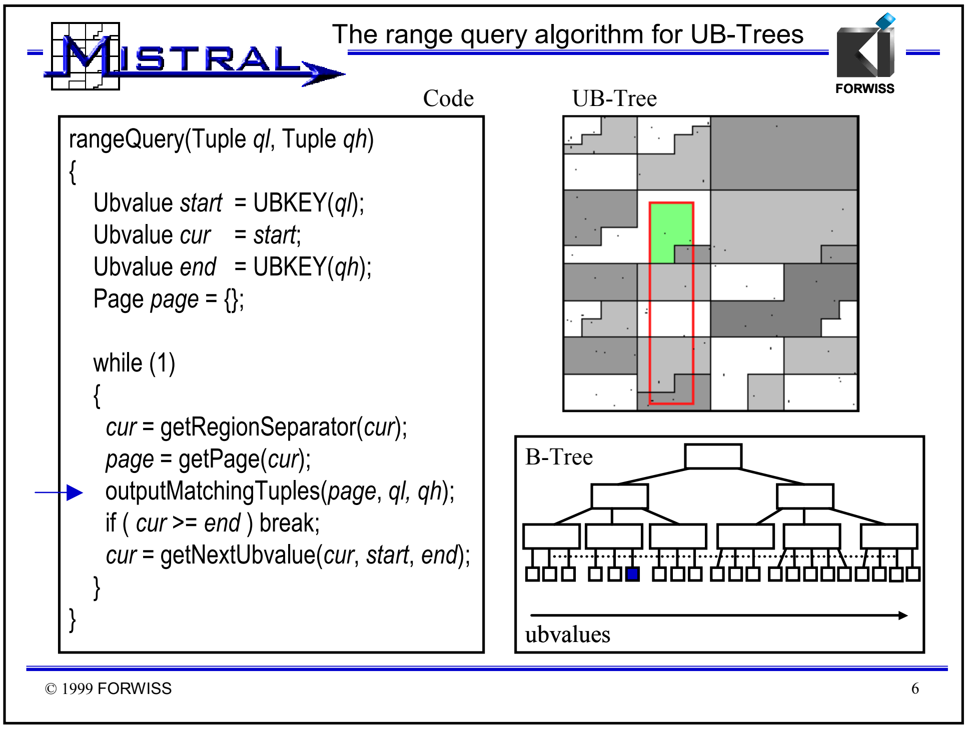
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
    Ubvalue start  = UBKEY(ql);
    Ubvalue cur    = start;
    Ubvalue end    = UBKEY(qh);
    Page page = {};

    while (1)
    {
      cur = getRegionSeparator(cur);
 ▶    page = getPage(cur);
      outputMatchingTuples(page, ql, qh);
      if ( cur >= end ) break;
      cur = getNextUbvalue(cur, start, end);
    }
}
```

© 1999 FORWISS 5



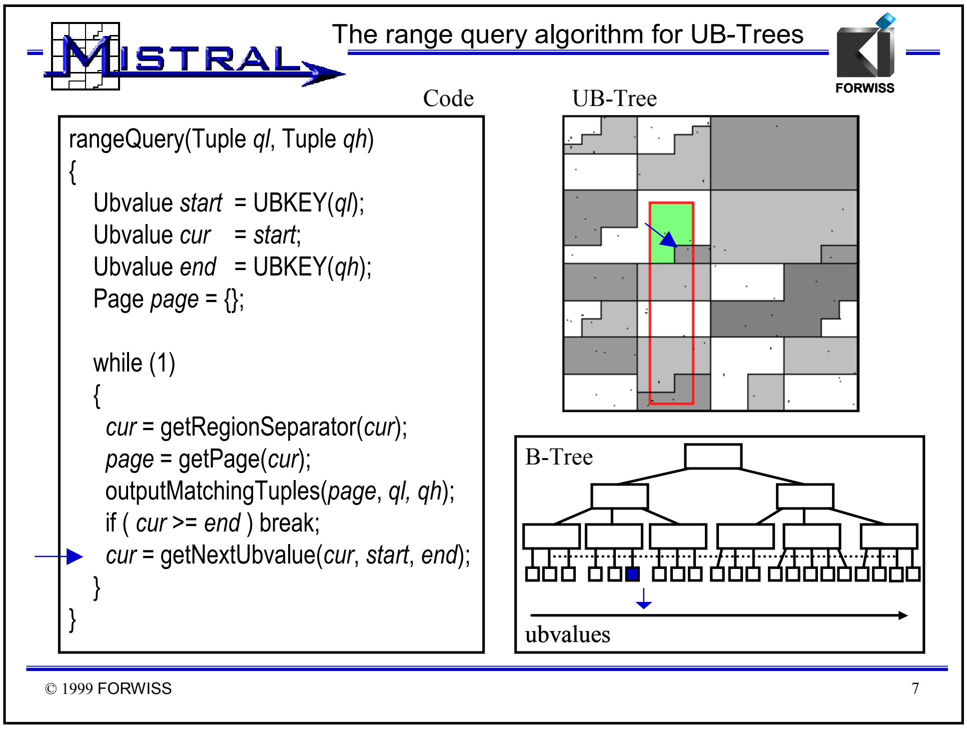
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
    Ubvalue start  = UBKEY(ql);
    Ubvalue cur    = start;
    Ubvalue end    = UBKEY(qh);
    Page page = {};

    while (1)
    {
      cur = getRegionSeparator(cur);
      page = getPage(cur);
 ▶    outputMatchingTuples(page, ql, qh);
      if ( cur >= end ) break;
      cur = getNextUbvalue(cur, start, end);
    }
}
```

© 1999 FORWISS 6



The range query algorithm for UB-Trees

FORWISS

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

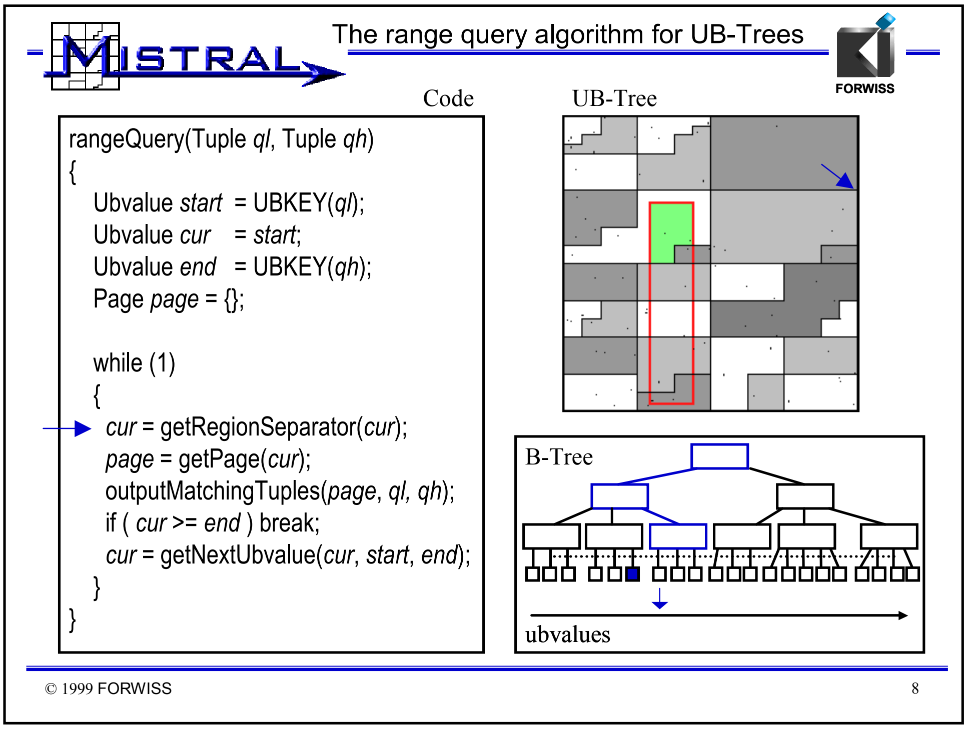
UB-Tree

B-Tree

ubvalues

© 1999 FORWISS

7



The range query algorithm for UB-Trees

FORWISS

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

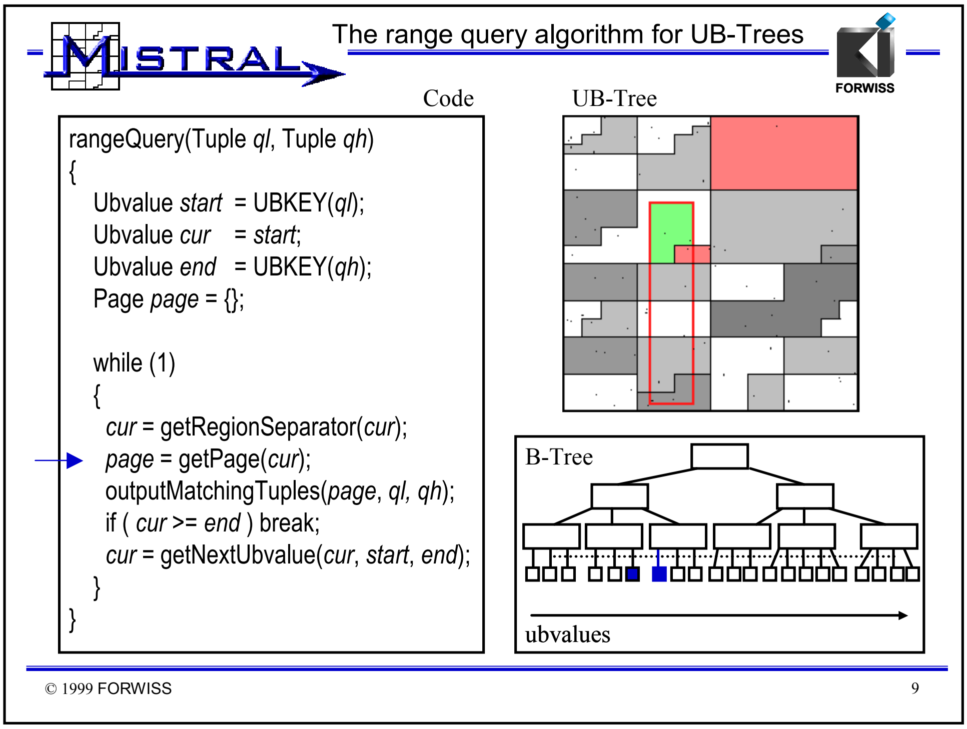
UB-Tree

B-Tree

ubvalues

© 1999 FORWISS

8

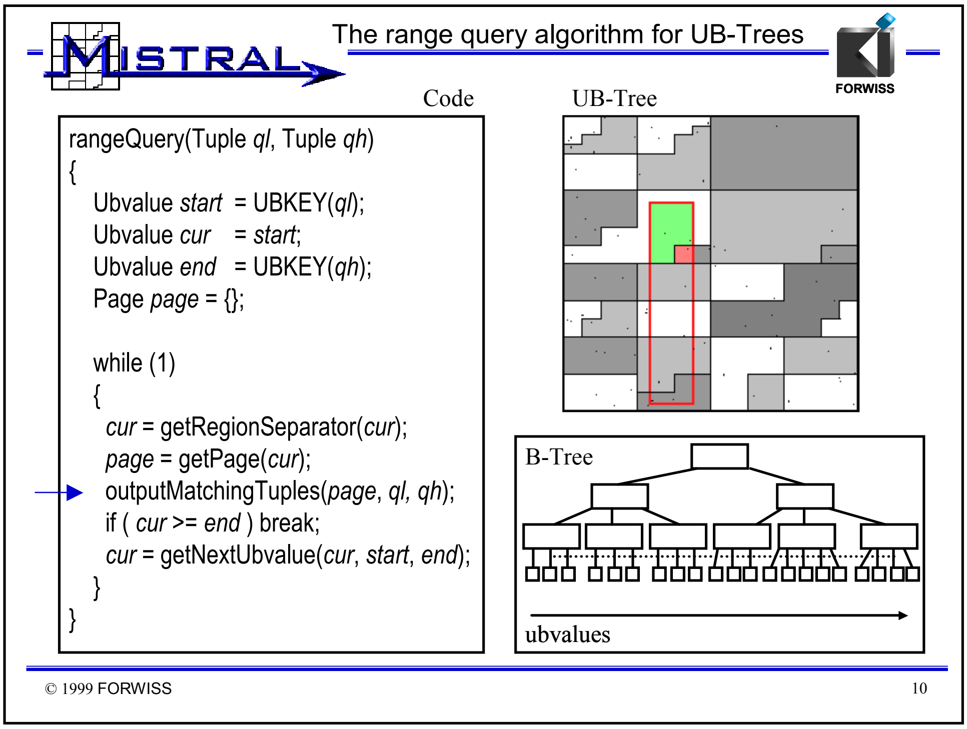


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS 9

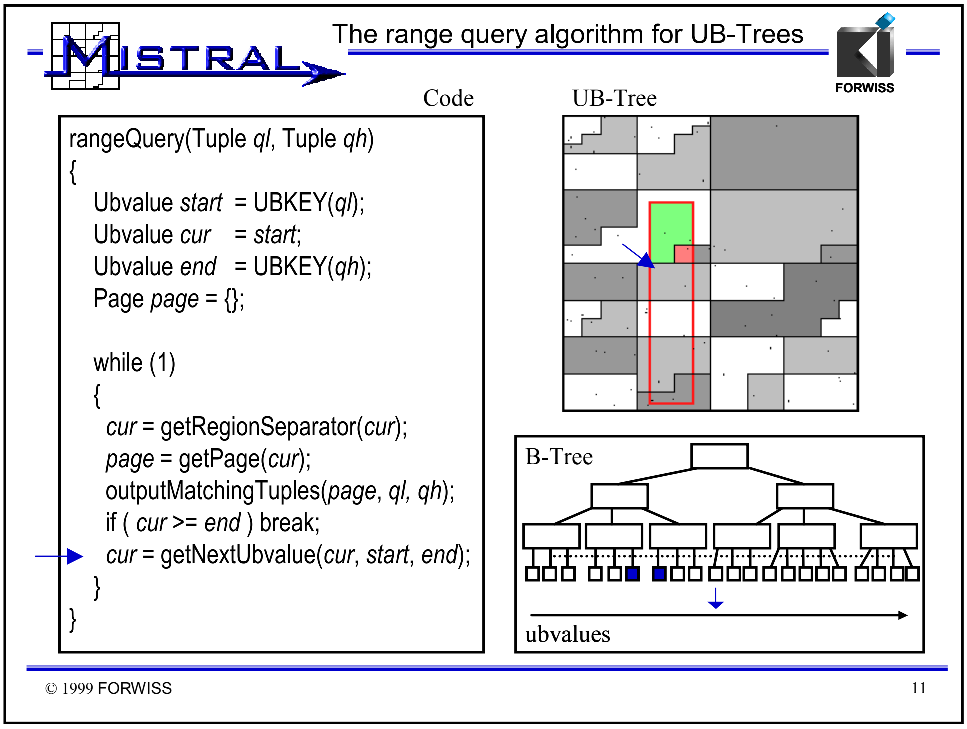


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS 10



The range query algorithm for UB-Trees

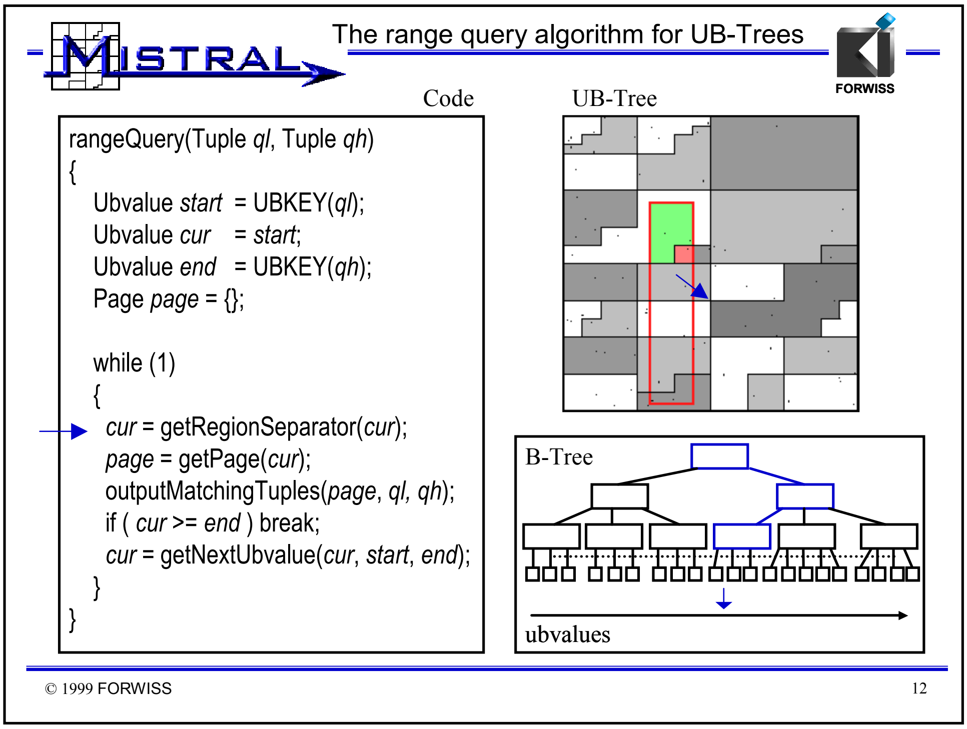
Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
→    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS

11



The range query algorithm for UB-Trees

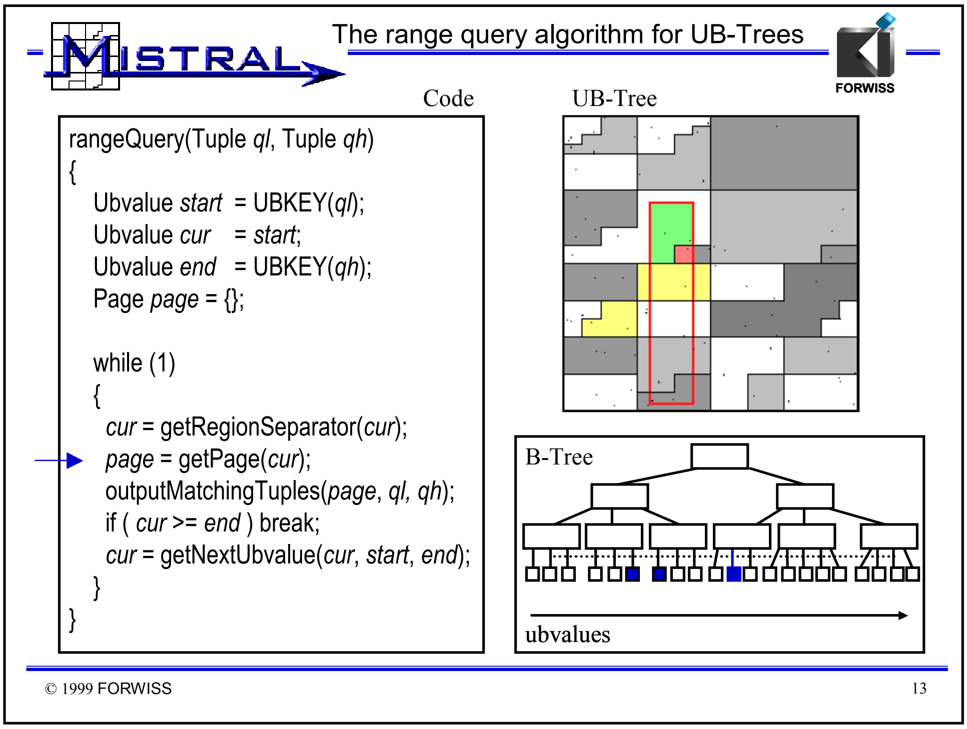
Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
→    cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS

12



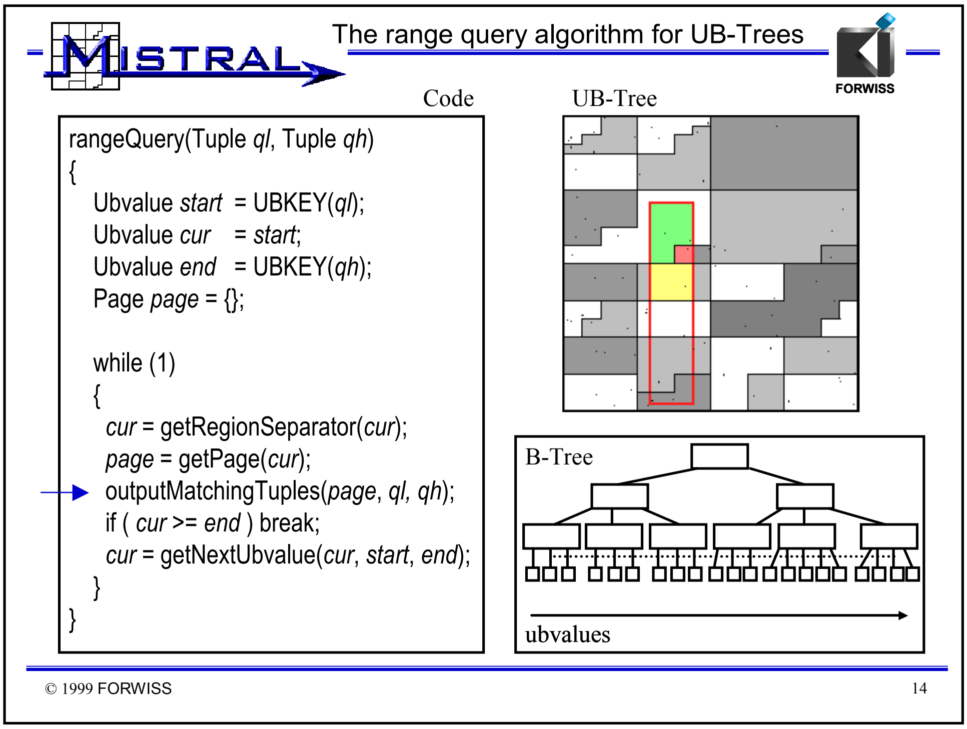
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
    Ubvalue start  = UBKEY(ql);
    Ubvalue cur    = start;
    Ubvalue end    = UBKEY(qh);
    Page page = {};

    while (1)
    {
      cur = getRegionSeparator(cur);
      page = getPage(cur);
      outputMatchingTuples(page, ql, qh);
      if ( cur >= end ) break;
      cur = getNextUbvalue(cur, start, end);
    }
}
```

© 1999 FORWISS 13



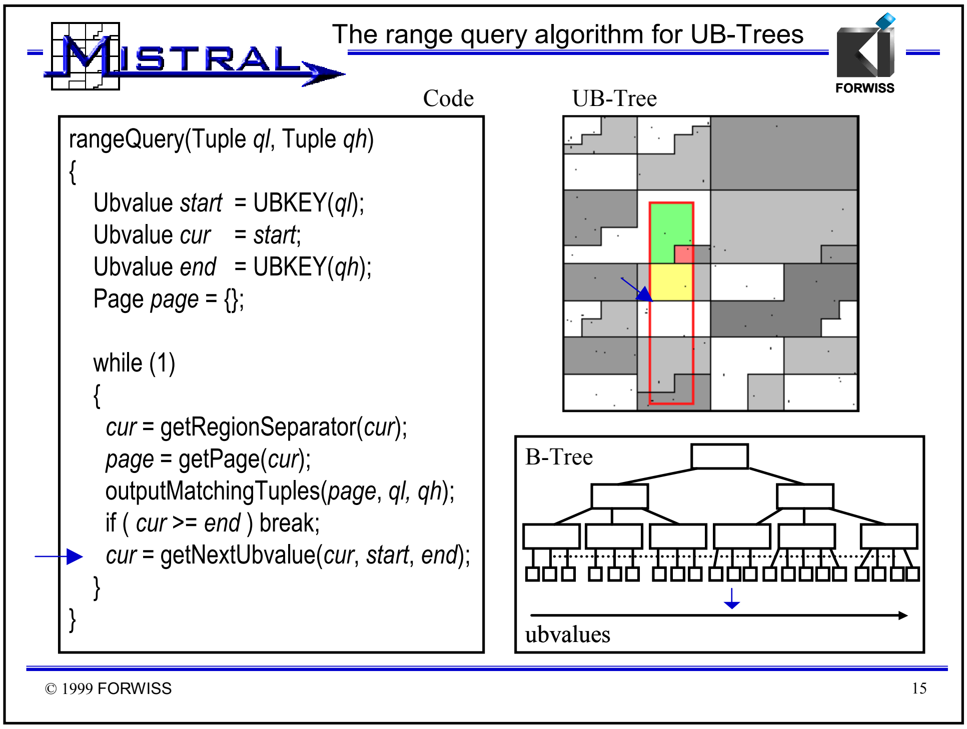
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
    Ubvalue start  = UBKEY(ql);
    Ubvalue cur    = start;
    Ubvalue end    = UBKEY(qh);
    Page page = {};

    while (1)
    {
      cur = getRegionSeparator(cur);
      page = getPage(cur);
      outputMatchingTuples(page, ql, qh);
      if ( cur >= end ) break;
      cur = getNextUbvalue(cur, start, end);
    }
}
```

© 1999 FORWISS 14

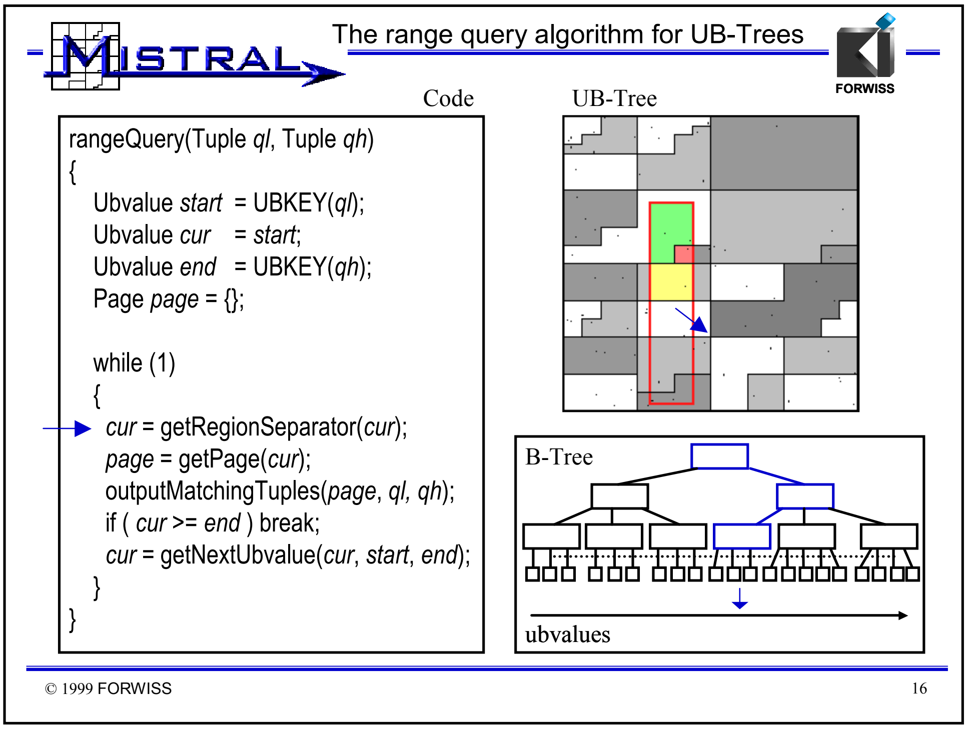


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
  Ubvalue start  = UBKEY(ql);
  Ubvalue cur    = start;
  Ubvalue end    = UBKEY(qh);
  Page page = {};

  while (1)
  {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
  }
}
```

© 1999 FORWISS 15

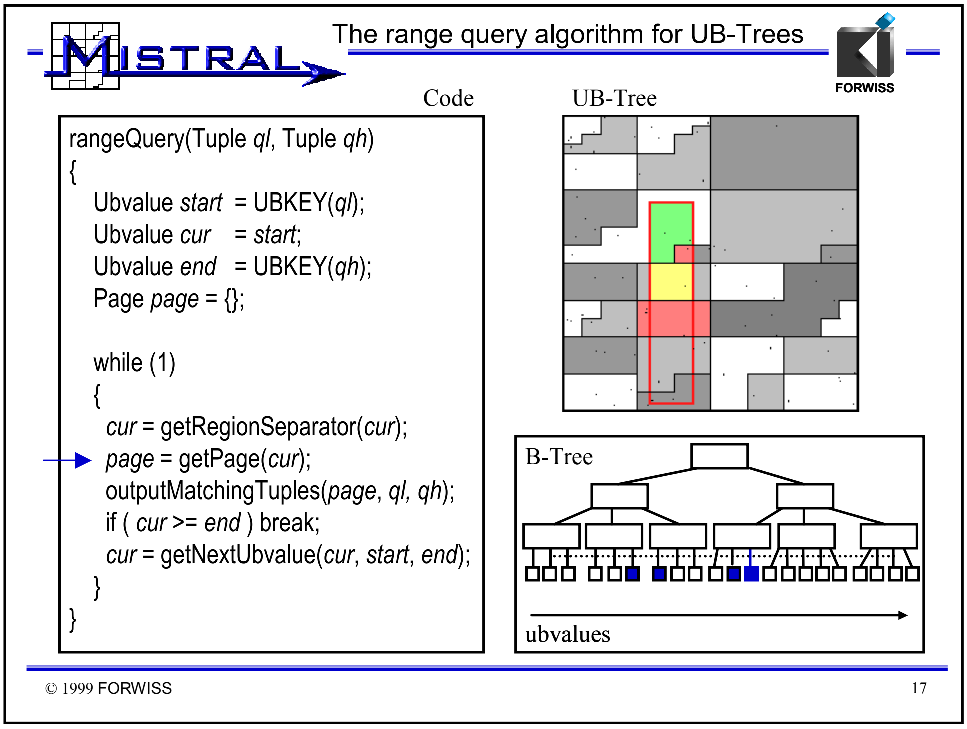


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
  Ubvalue start  = UBKEY(ql);
  Ubvalue cur    = start;
  Ubvalue end    = UBKEY(qh);
  Page page = {};

  while (1)
  {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
  }
}
```

© 1999 FORWISS 16

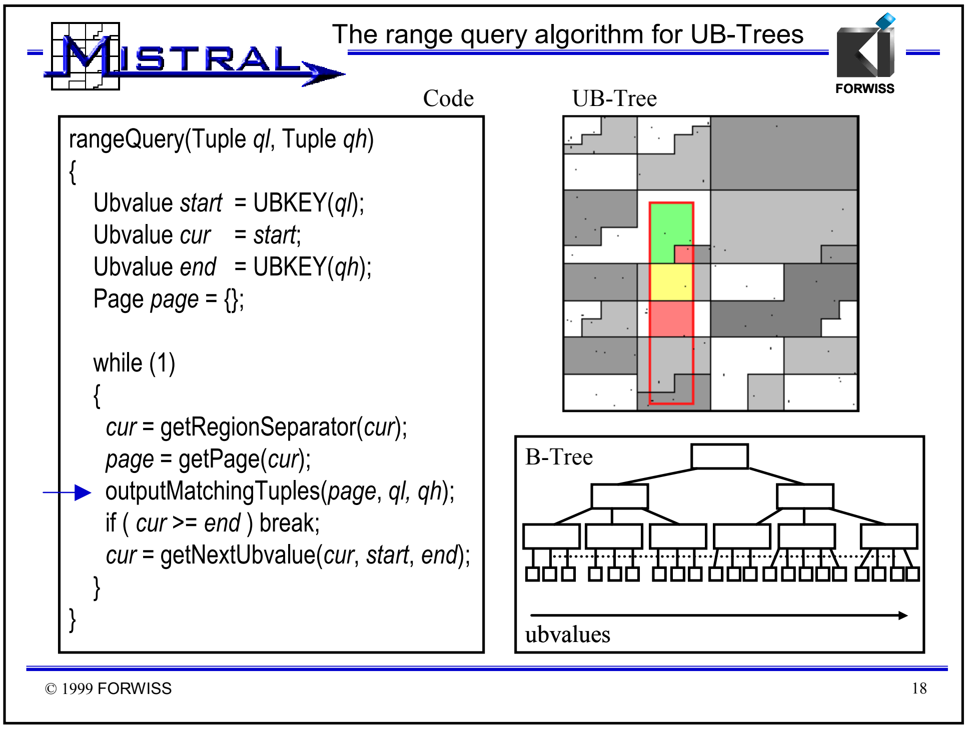


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
→    page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

17

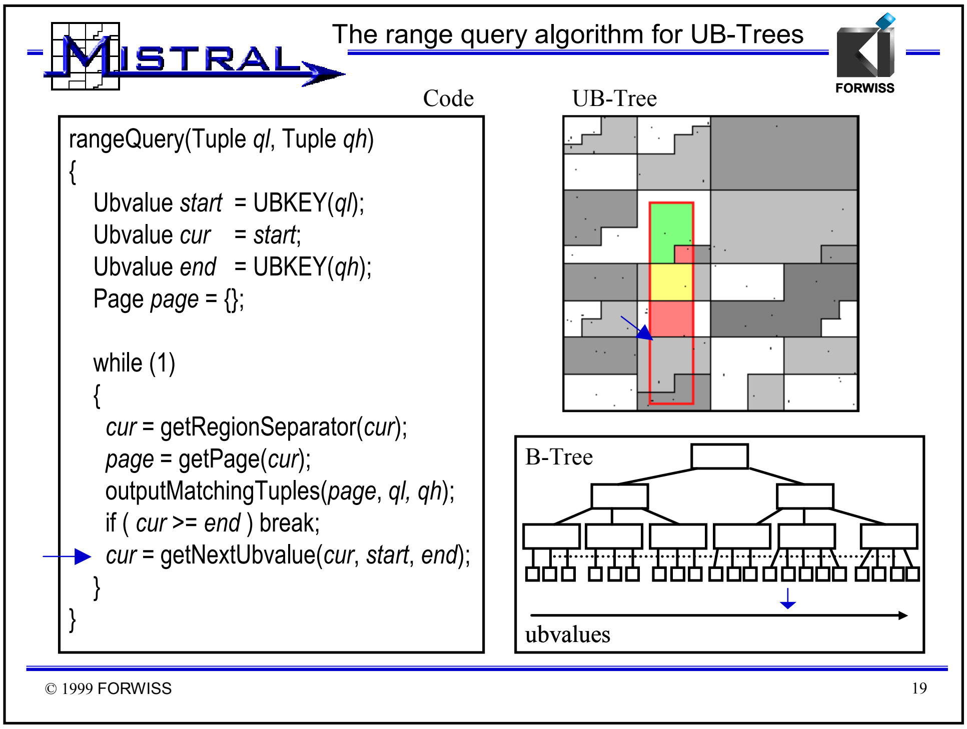


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
→    outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

18



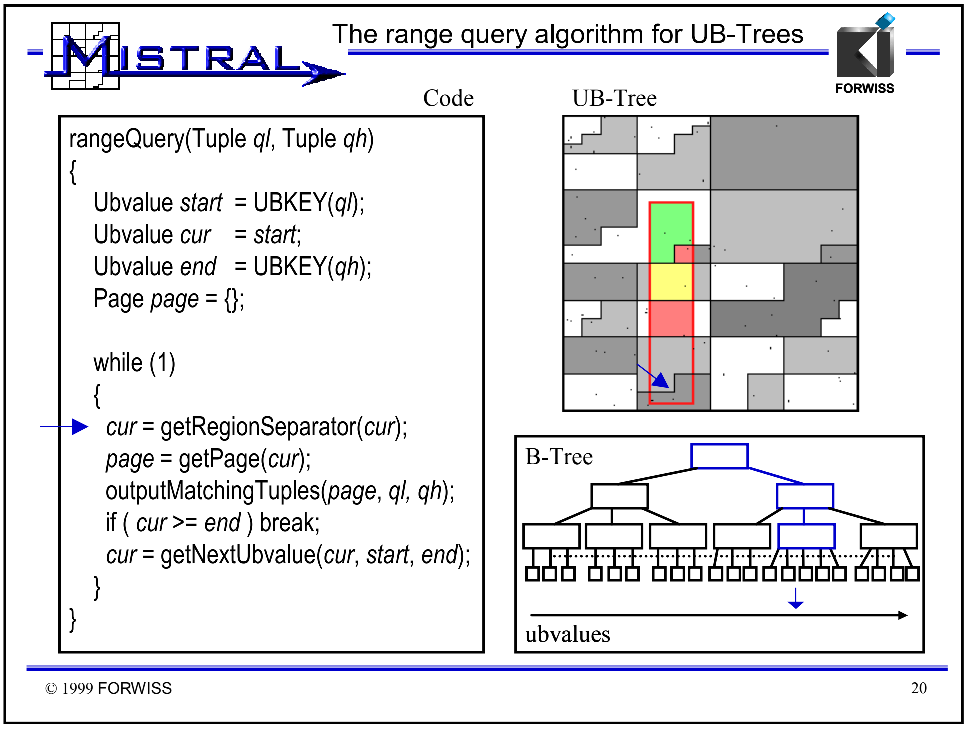
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
→    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS 19



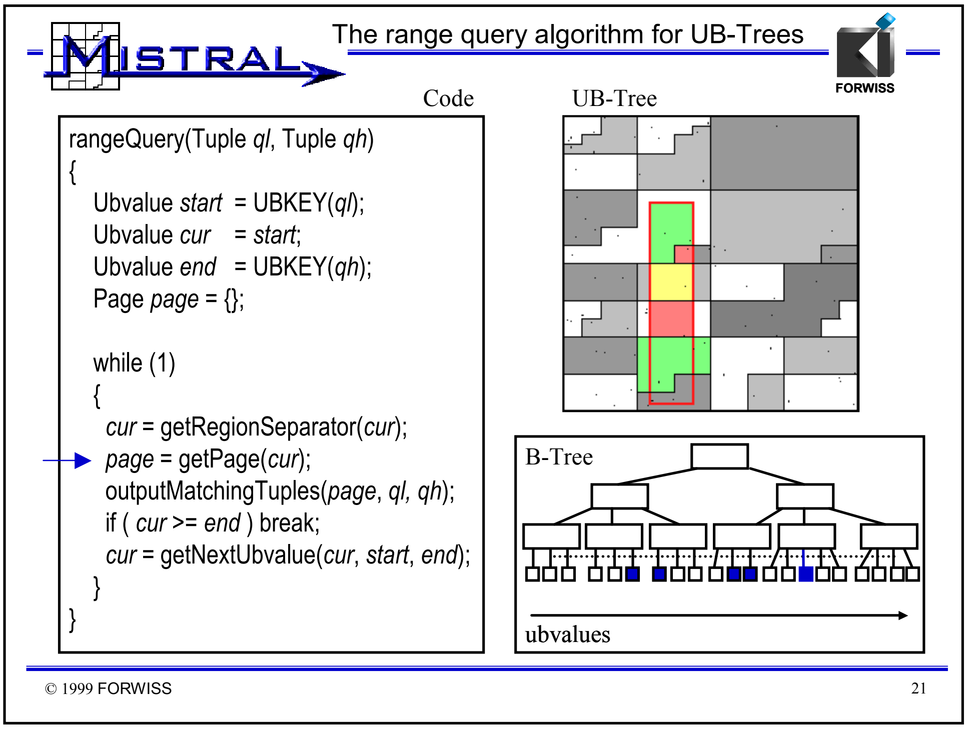
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
→    cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS 20



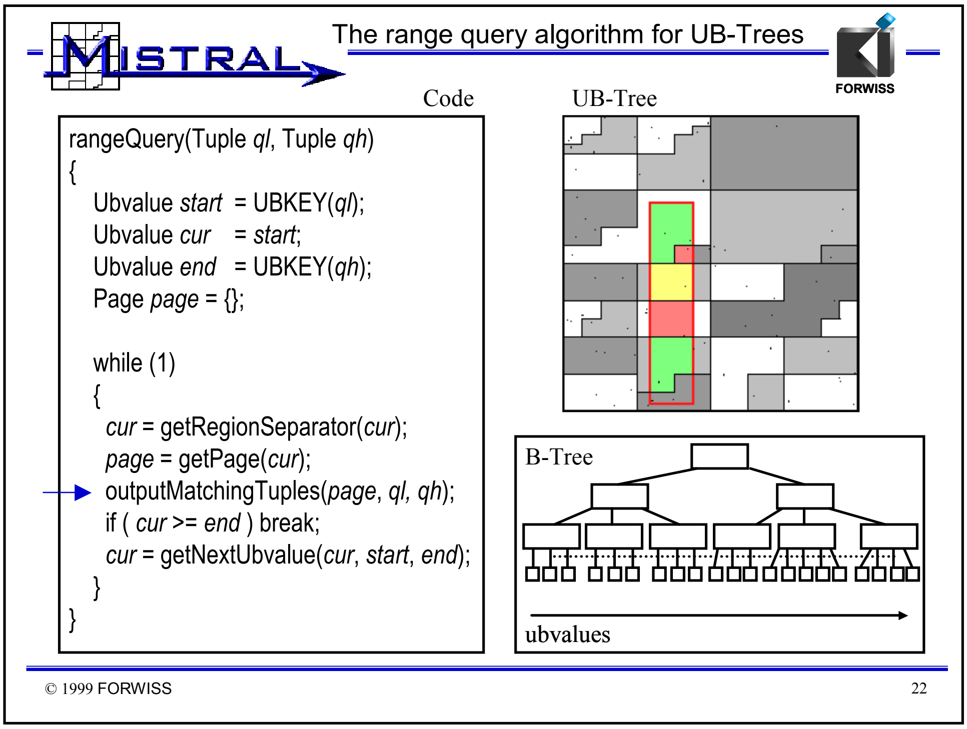
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
→   page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

21



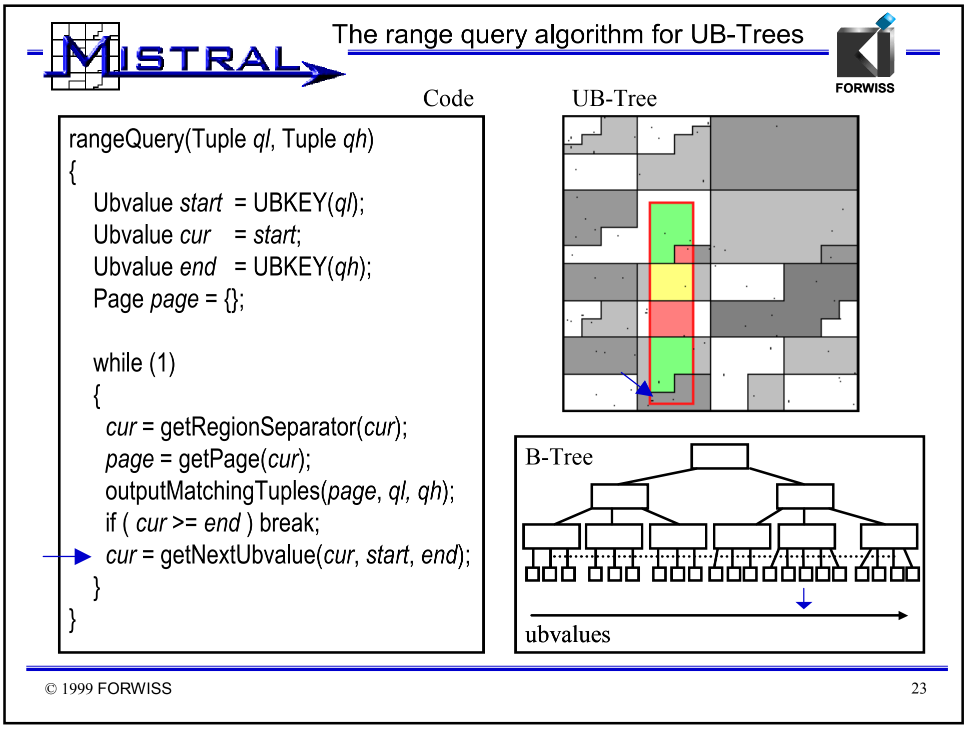
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
→   outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

22



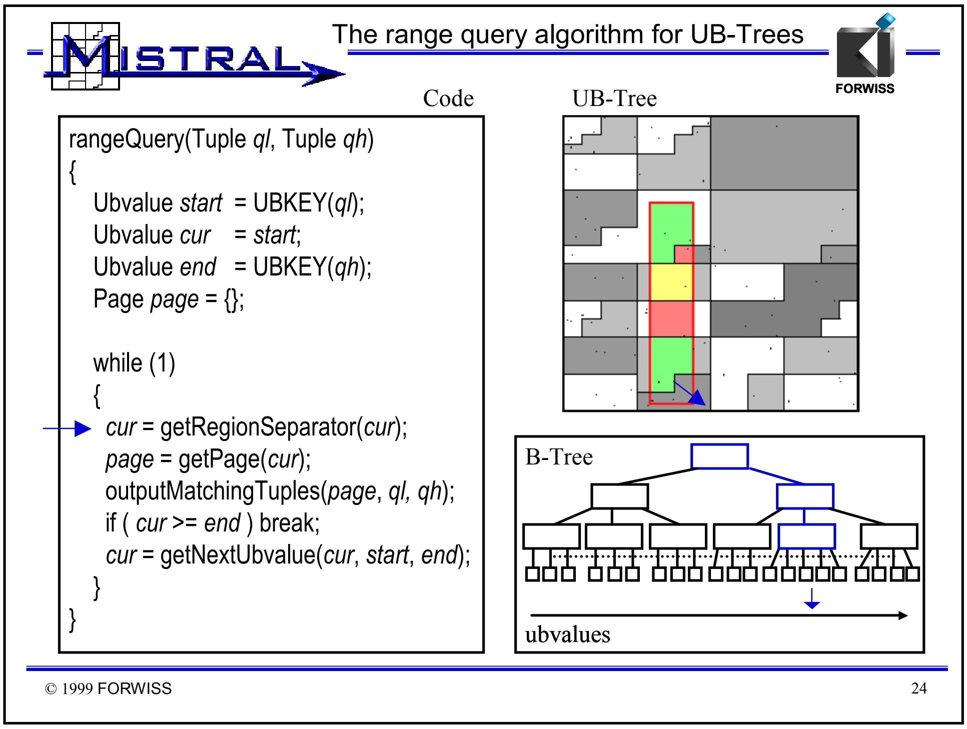
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
→   cur = getNextUbvalue(cur, start, end);
   }
}
```

23



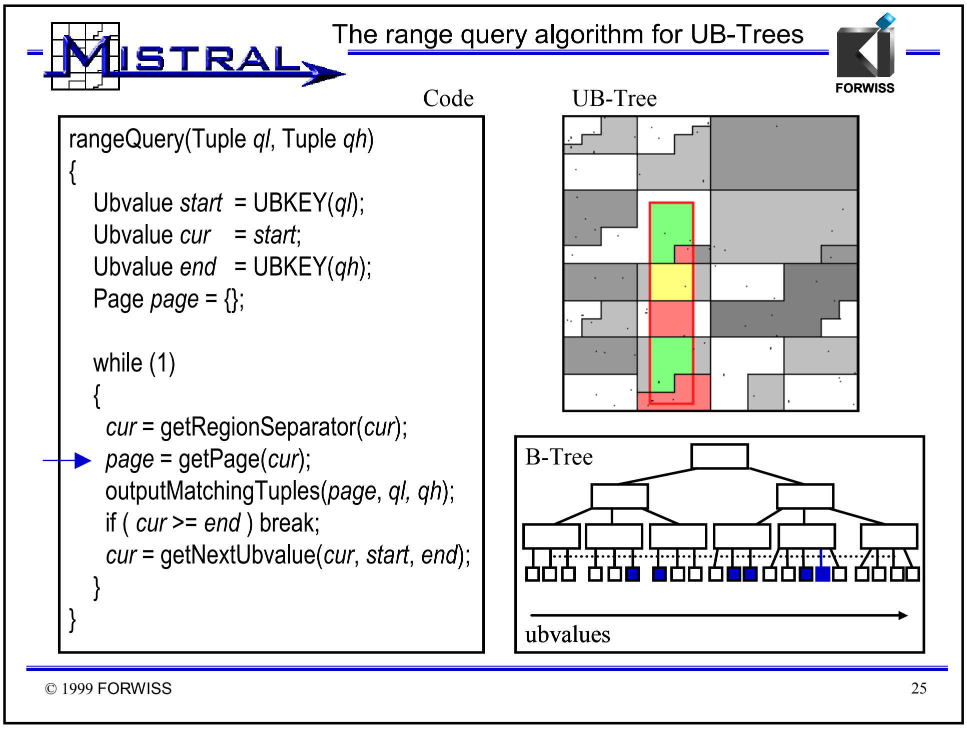
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
→   cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

24



The range query algorithm for UB-Trees

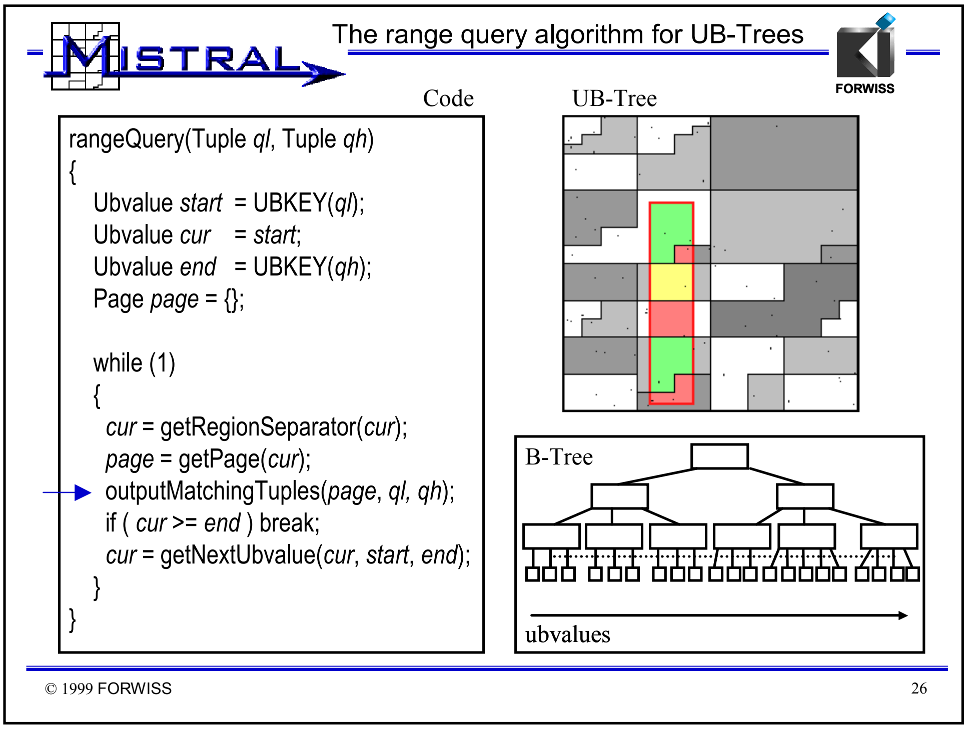
Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS

25



The range query algorithm for UB-Trees

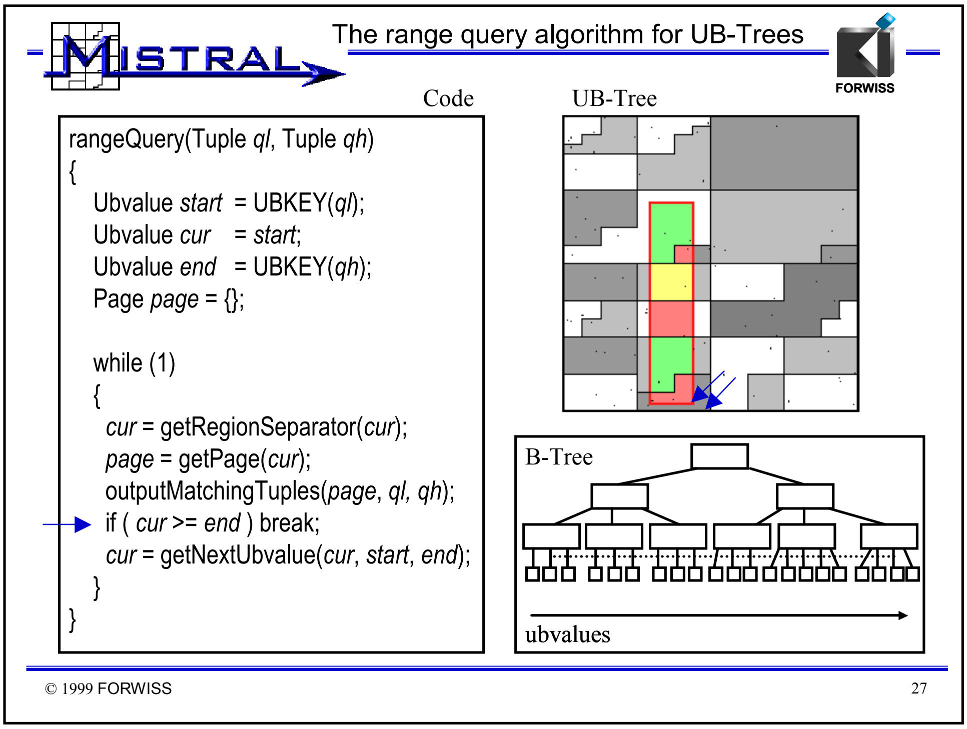
Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS

26

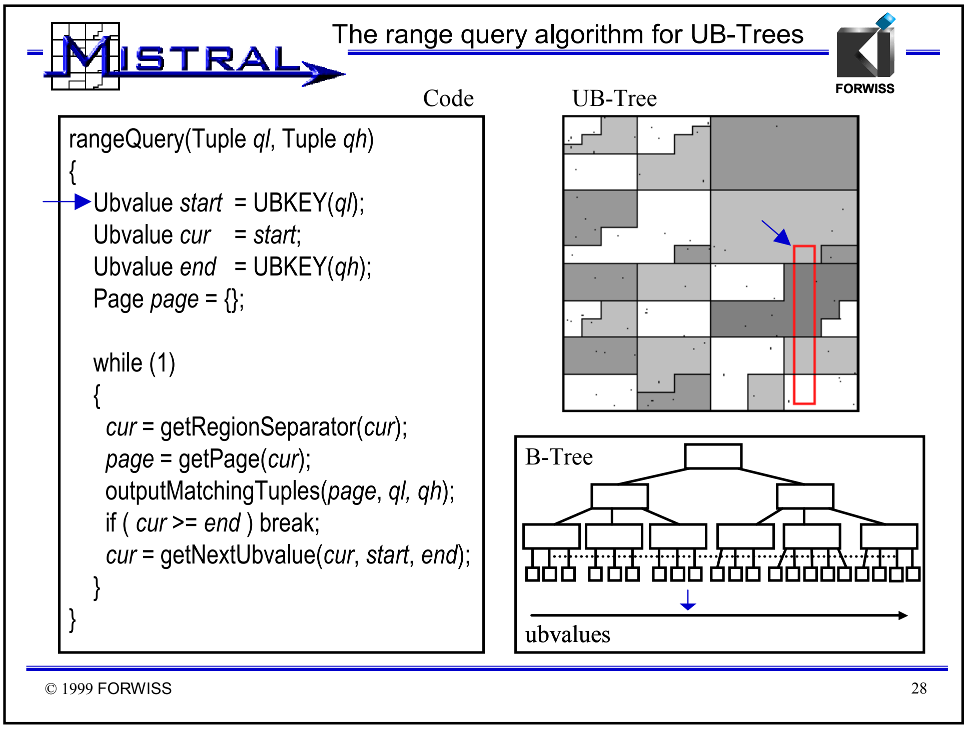


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
  Ubvalue start  = UBKEY(ql);
  Ubvalue cur    = start;
  Ubvalue end    = UBKEY(qh);
  Page page = {};

  while (1)
  {
   cur = getRegionSeparator(cur);
   page = getPage(cur);
   outputMatchingTuples(page, ql, qh);
→  if ( cur >= end ) break;
   cur = getNextUbvalue(cur, start, end);
  }
}
```

© 1999 FORWISS 27

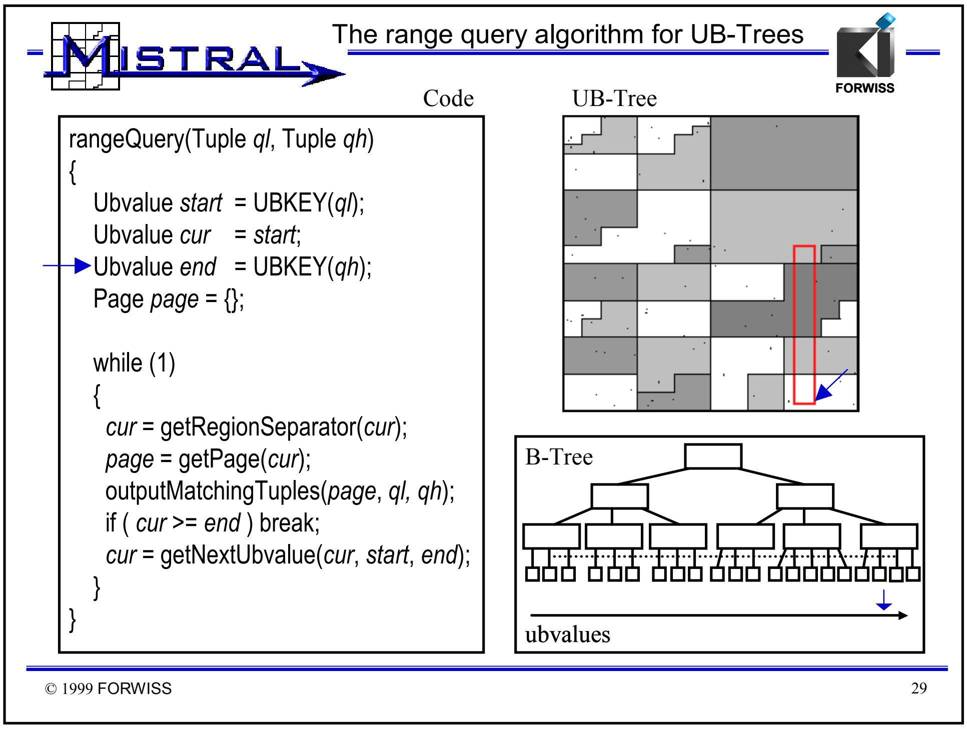


The range query algorithm for UB-Trees

```
rangeQuery(Tuple ql, Tuple qh)
{
→ Ubvalue start  = UBKEY(ql);
  Ubvalue cur    = start;
  Ubvalue end    = UBKEY(qh);
  Page page = {};

  while (1)
  {
   cur = getRegionSeparator(cur);
   page = getPage(cur);
   outputMatchingTuples(page, ql, qh);
   if ( cur >= end ) break;
   cur = getNextUbvalue(cur, start, end);
  }
}
```

© 1999 FORWISS 28



MISTRAL

The range query algorithm for UB-Trees

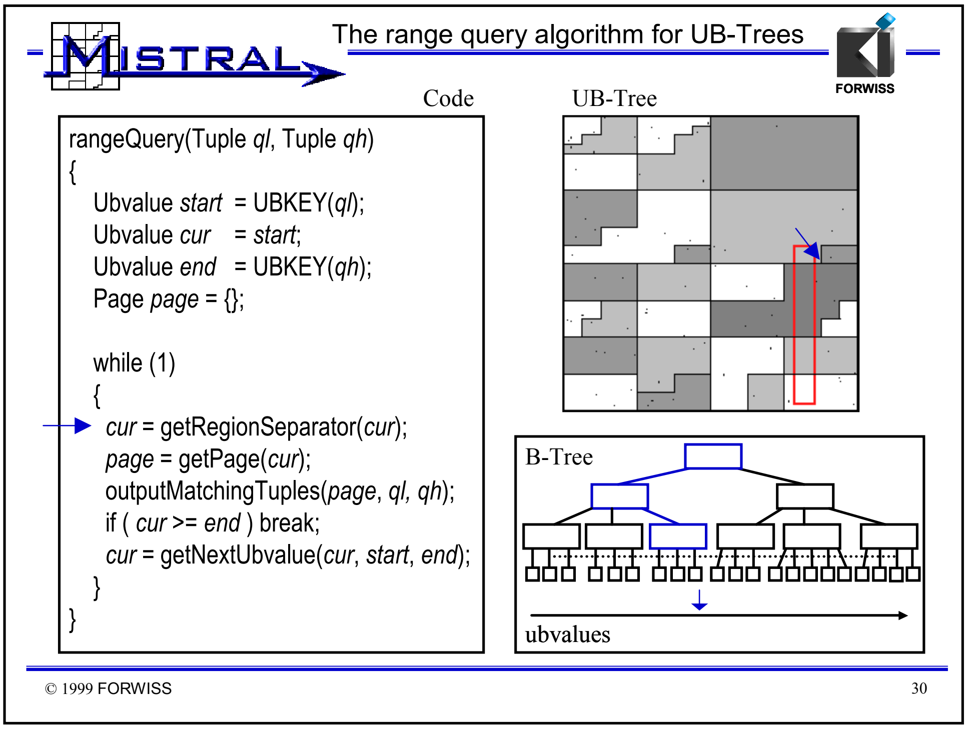
FORWISS

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS 29



MISTRAL

The range query algorithm for UB-Trees

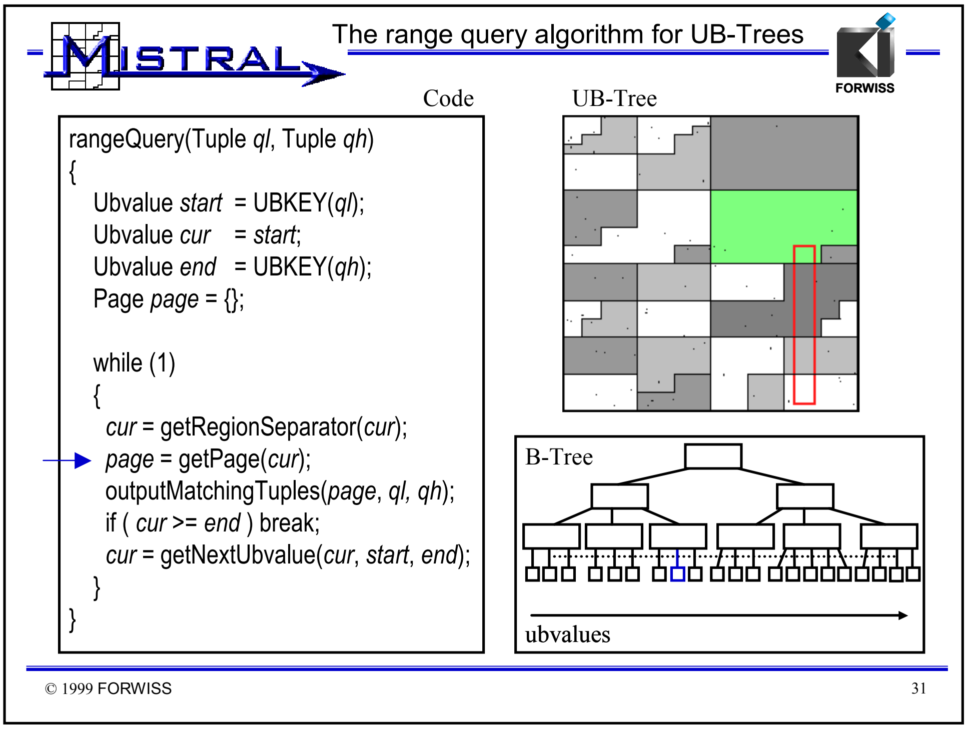
FORWISS

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS 30



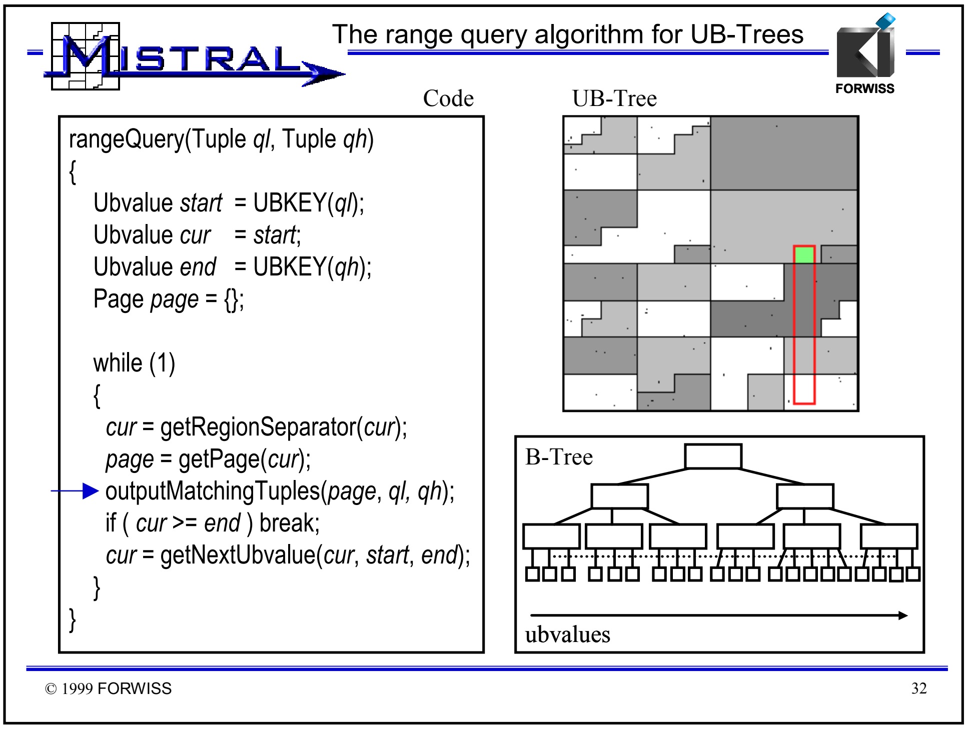
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS



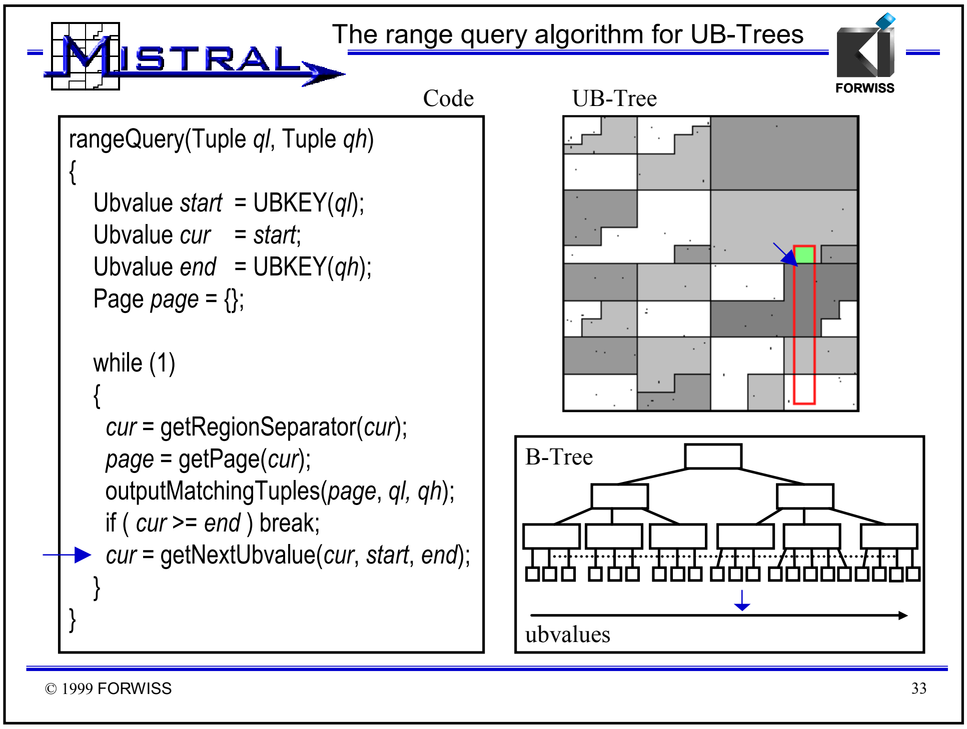
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS



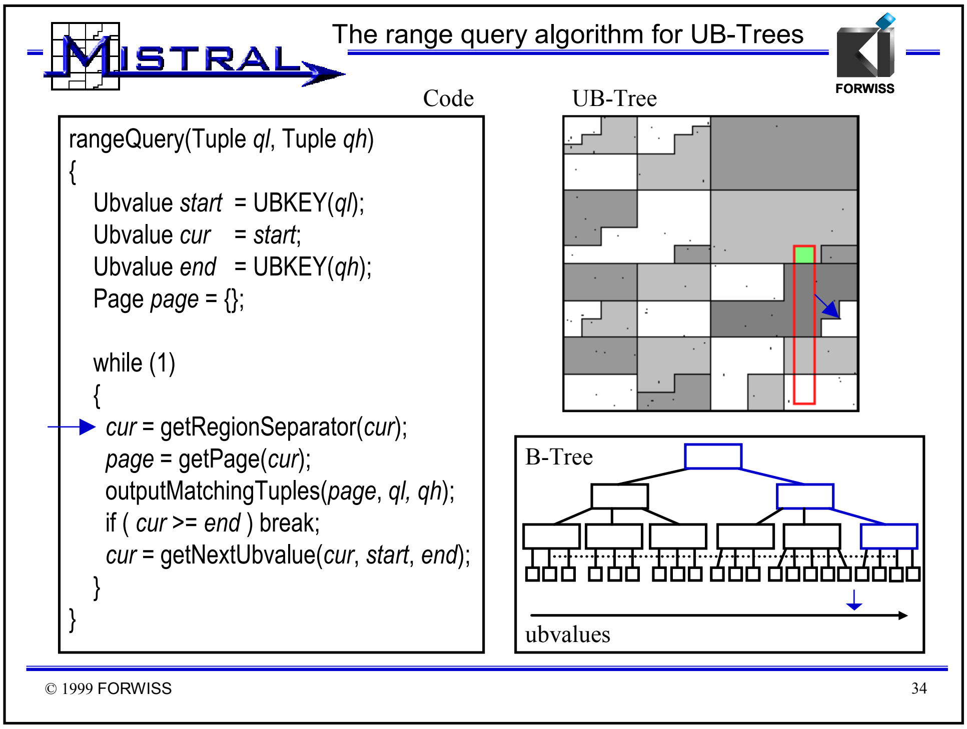
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS 33



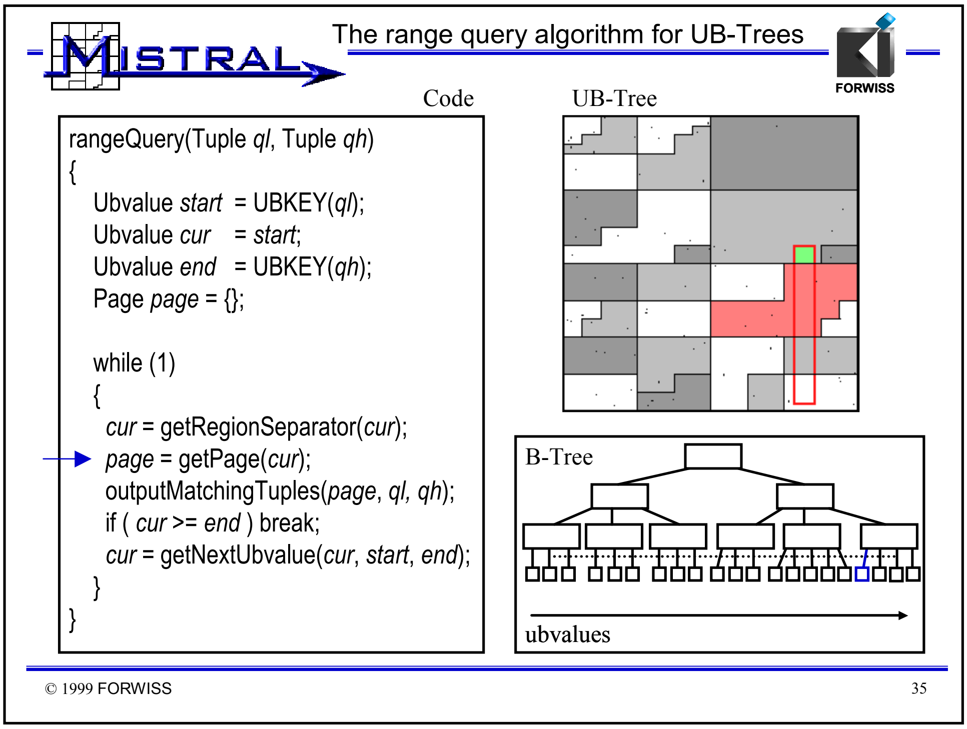
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS 34



The range query algorithm for UB-Trees

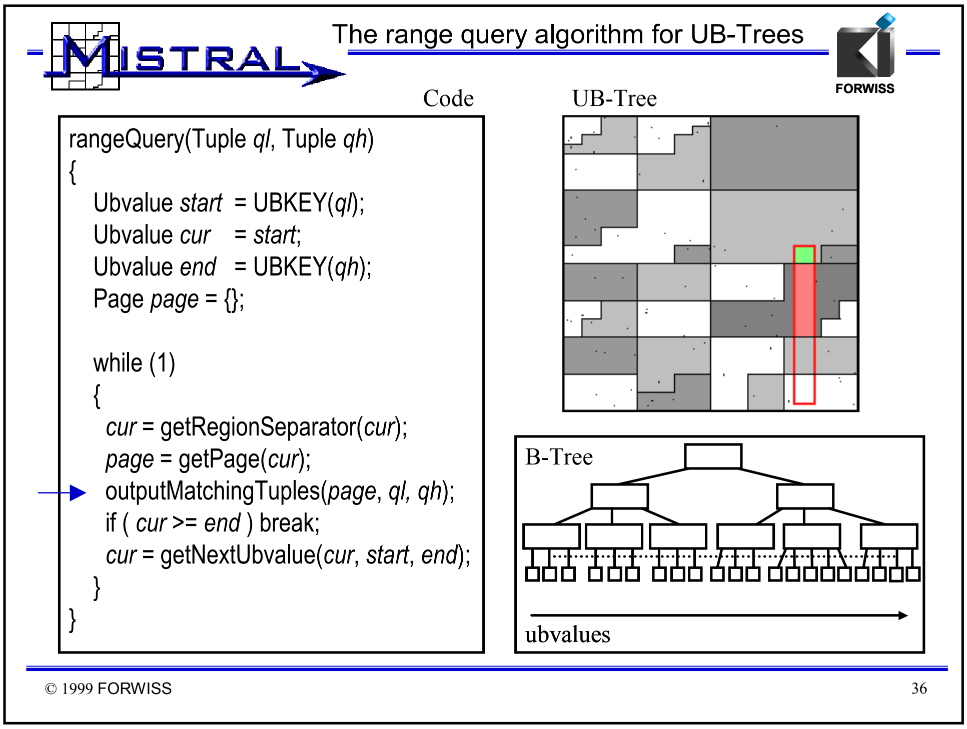
Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
→    page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS

35



The range query algorithm for UB-Trees

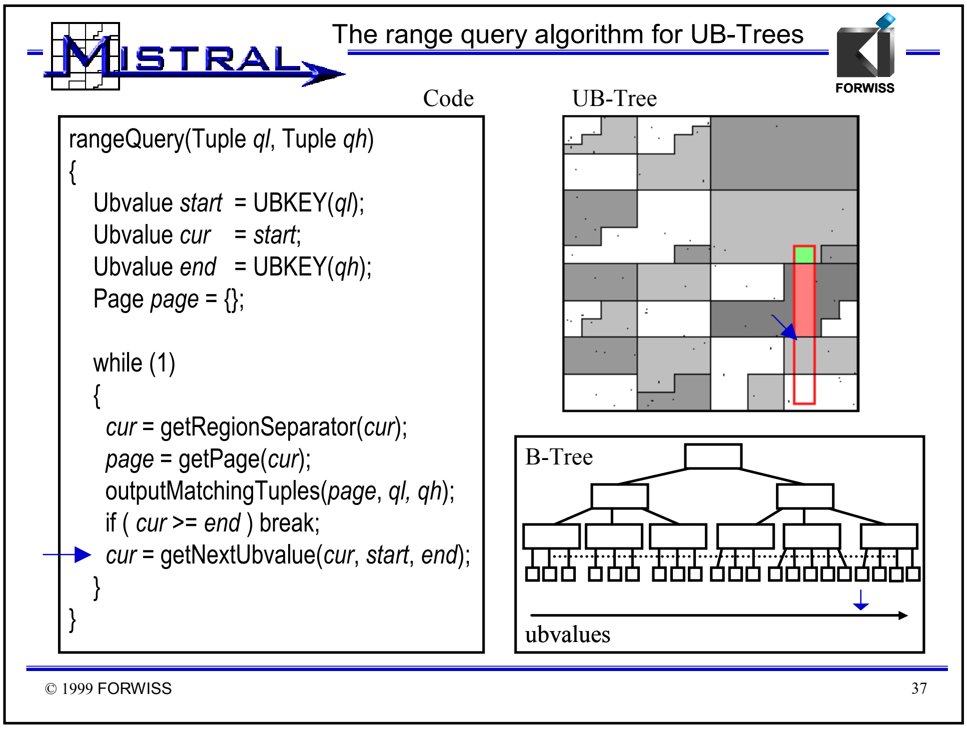
Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
→    outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS

36



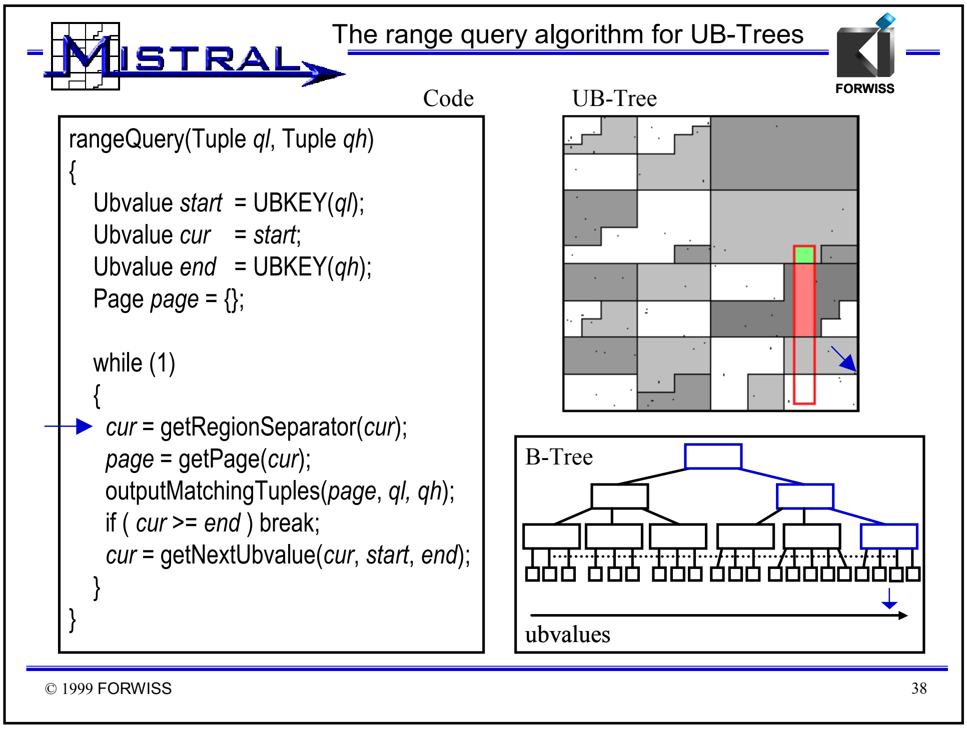
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
→    cur = getNextUbvalue(cur, start, end);
   }
}
```

37



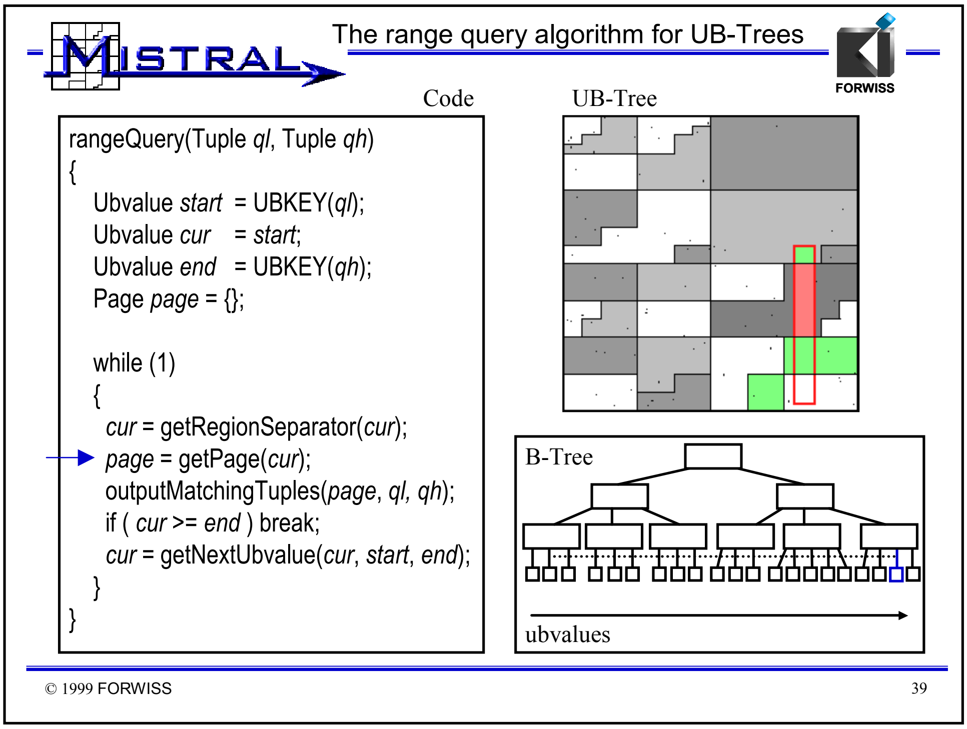
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
→    cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

38



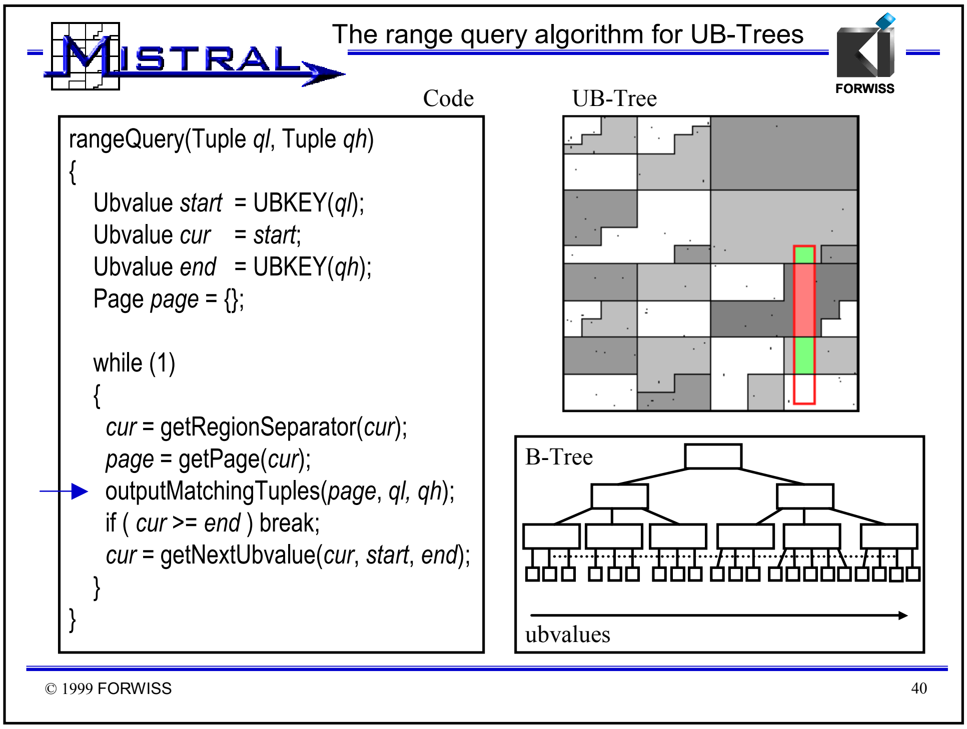
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
    Ubvalue start  = UBKEY(ql);
    Ubvalue cur    = start;
    Ubvalue end    = UBKEY(qh);
    Page page = {};

    while (1)
    {
      cur = getRegionSeparator(cur);
      page = getPage(cur);
      outputMatchingTuples(page, ql, qh);
      if ( cur >= end ) break;
      cur = getNextUbvalue(cur, start, end);
    }
}
```

39



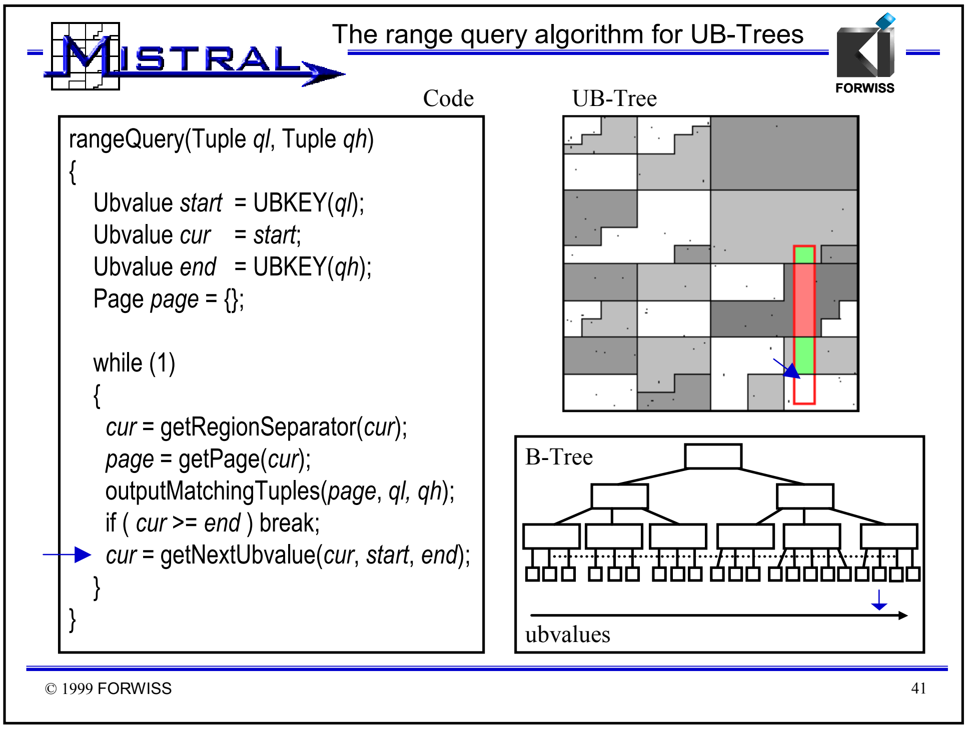
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
    Ubvalue start  = UBKEY(ql);
    Ubvalue cur    = start;
    Ubvalue end    = UBKEY(qh);
    Page page = {};

    while (1)
    {
      cur = getRegionSeparator(cur);
      page = getPage(cur);
      outputMatchingTuples(page, ql, qh);
      if ( cur >= end ) break;
      cur = getNextUbvalue(cur, start, end);
    }
}
```

40



The range query algorithm for UB-Trees

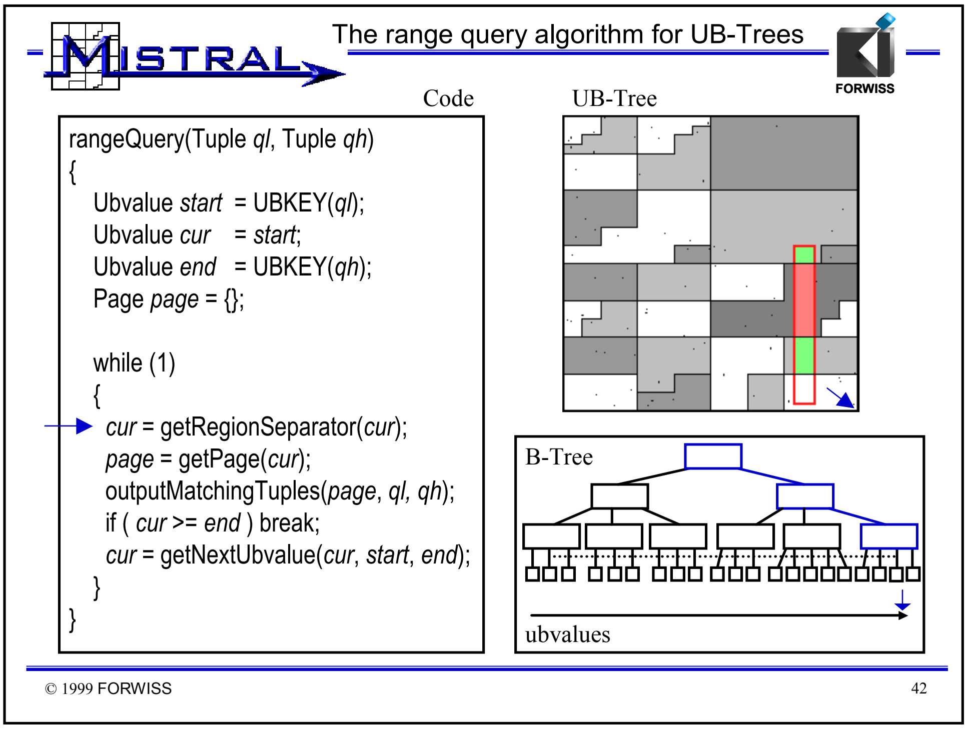
Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS

41



The range query algorithm for UB-Trees

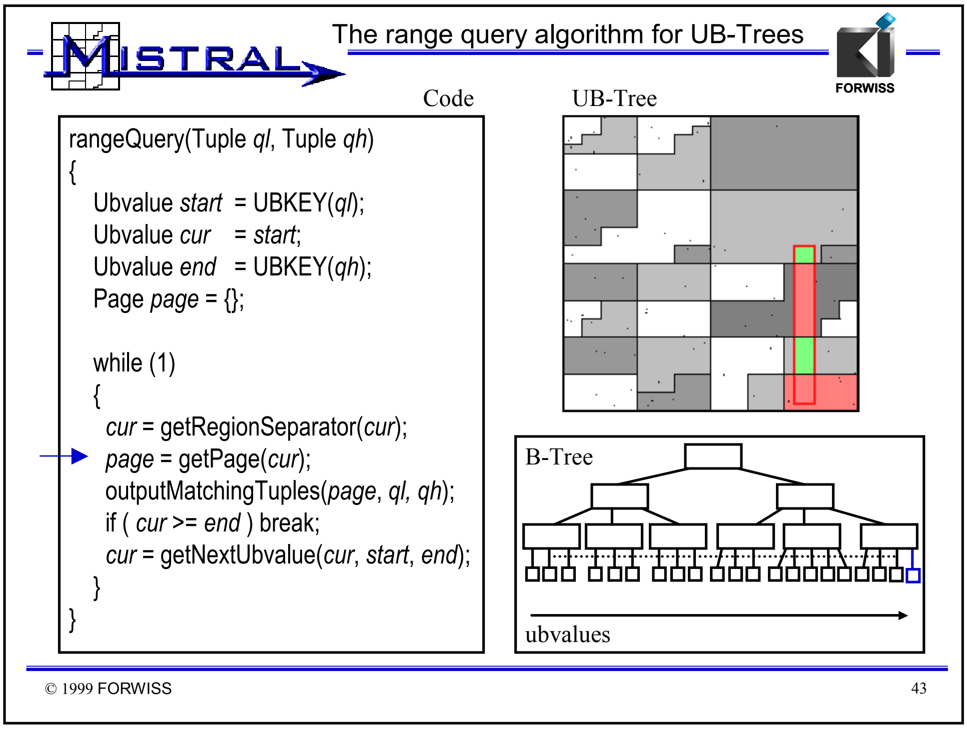
Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
    cur = getRegionSeparator(cur);
    page = getPage(cur);
    outputMatchingTuples(page, ql, qh);
    if ( cur >= end ) break;
    cur = getNextUbvalue(cur, start, end);
   }
}
```

© 1999 FORWISS

42



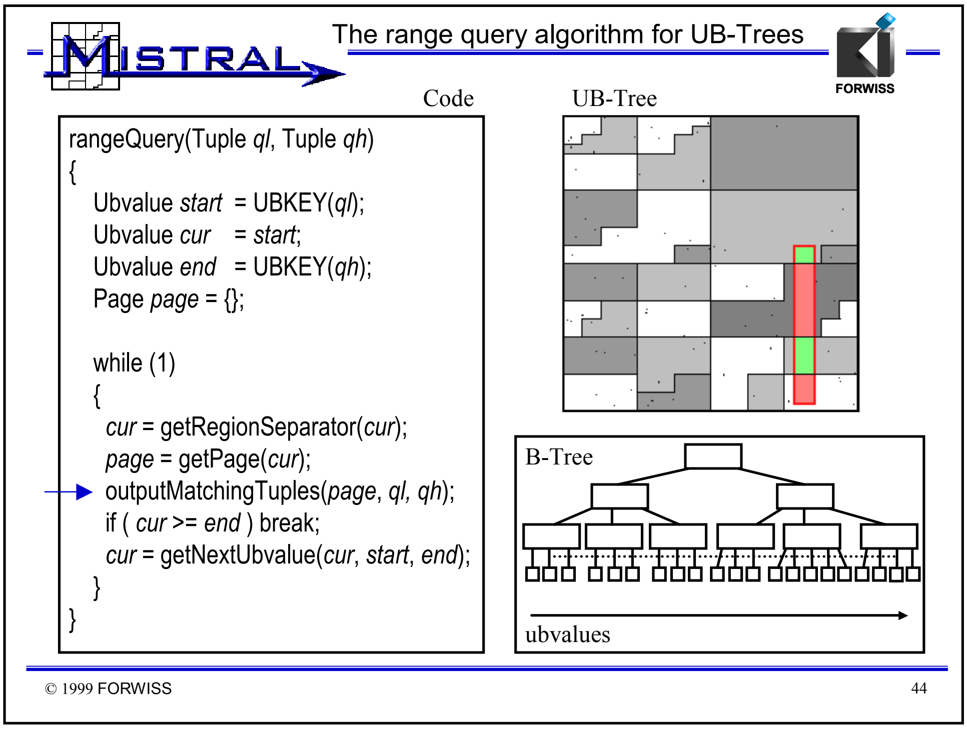
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
→    page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

43



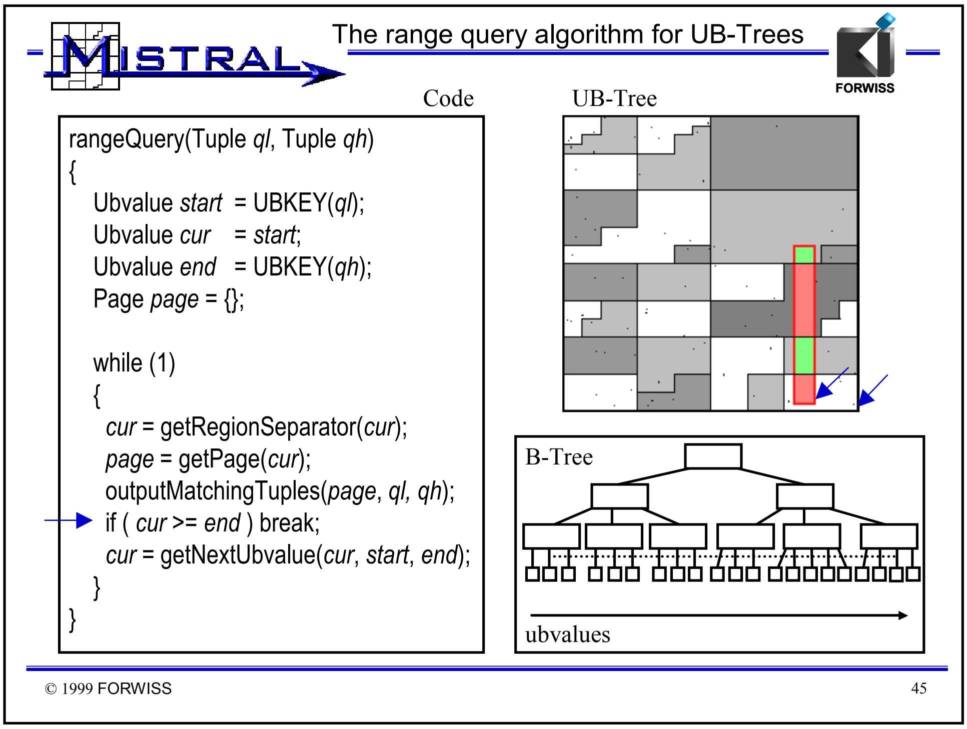
The range query algorithm for UB-Trees

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
→    outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

44



MISTRAL

The range query algorithm for UB-Trees

FORWISS

Code

```
rangeQuery(Tuple ql, Tuple qh)
{
   Ubvalue start  = UBKEY(ql);
   Ubvalue cur    = start;
   Ubvalue end    = UBKEY(qh);
   Page page = {};

   while (1)
   {
     cur = getRegionSeparator(cur);
     page = getPage(cur);
     outputMatchingTuples(page, ql, qh);
     if ( cur >= end ) break;
     cur = getNextUbvalue(cur, start, end);
   }
}
```

UB-Tree